Large Eddy Simulation of Internal Combustion Engine with Efficient, Particle-Based Moving Immersed Boundaries

Von der Fakultät für Ingenieurwissenschaften, Abteilung Maschinenbau und Verfahrenstechnik

der

Universität Duisburg-Essen

zur Erlangung des akademischen Grades

eines

Doktors der Ingenieurwissenschaften

Dr.-Ing.

genehmigte Dissertation

von

Thuong Nguyen
aus
Langson - Vietnam

Gutachter: Univ.-Prof. Dr.-Ing. Andreas Kempf
Univ.-Prof. Dr.-Ing. habil. Markus Klein
Tag der mündlichen Prüfung: 17.07.2017

Declaration of Authorship

I, Thuong Nguyen, declare that this thesis titled, 'Large Eddy Simulation of Internal Combustion Engine with Efficient, Particle-Based Moving Immersed Boundaries' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

"The important thing is not to stop questioning. Curiosity has its own reason for existing. One cannot help but be in awe when he contemplates the mysteries of eternity, of life, of the marvelous structure of reality."

Albert Einstein

Abstract

Numerical simulations play an important role in research and development of internal combustion engines (ICE). Existing numerical approaches for engine simulation have been proven to be accurate in reproducing a wide range of physical phenomena inside an ICE. Numerical studies on more complex issues related to ICE, such as NOx formation, knock, ignition or cycle-to-cycle variation (CCV), have become more feasible and reliable. Together with the increase in the computational capacity and the advancement in the numerical methods, developing numerical approaches that are easy to implement and efficient to utilize the high performance computing (HPC) become more demanding.

This work focuses on the mathematical modelling and the numerical simulations to study the flow dynamics and the combustion processes inside an ICE. The aim is to develop simple but efficient approaches to tackle a number of issues that are important for the ICE simulations. This dissertation proposes a new approach for the engine simulations based on the immersed boundary method (IBM), Lagrangian particles and large eddy simulation (LES). In this work, an LES solver for a fully compressible Navier-Stokes equations is implemented to simulate the flow dynamics inside an ICE. To model the interaction between the fluid flow and the moving boundaries, a particle-based method is developed and implemented into the compressible flow solver. Applying this method to the engine simulations, the motion of the valves and the piston is numerically described by Lagrangian particles. This proposed mesh-free technique for boundary representation is simple for parallelization and suitable for high performance computing. The stationary parts of the engine are described by a computationally efficient IBM, where the mesh generation is fast and simple on an equidistant Cartesian grid, and the involved numerical operations can be efficiently performed by vectorization. In the proposed approach, the turbulent combustion inside the engine combustion chamber is modelled by the flame surface density (FSD) approach using a single step chemical mechanism for the heat release.

To demonstrate the method, LES results are presented for different test cases involving the flow around a cylinder, the flow in channels, the motored- and fired-case in different engine geometries. Good agreements are achieved between the numerical results and the experiments. The obtained results show the suitability and the reliability of the proposed method for accurate numerical predictions in various CFD-problems. The suitability of the proposed approach for HPC is demonstrated by massively parallel simulations of an ICE on different supercomputers.

Zusammenfassung

Numerische Simulationen spielen eine wichtige Rolle bei der Forschung und Entwicklung von Verbrennungsmotoren. Bestehende numerische Ansätze für die Simulation von Motoren haben sich als präzise erwiesen, um eine breite Palette von physikalischen Phänomenen innerhalb eines Verbrennungsmotors zu reproduzieren. Numerische Studien zu komplexeren Fragen im Zusammenhang mit Verbrennungsmotoren, wie z.B. NOx-Bildung, Klopfen, Zündung oder Zyklus-zu-Zyklus-Variation, sind umsetzbarer und zuverlässiger geworden. Die Entwicklung von numerischen Methoden, die einfach zu implementieren sind und High-Performance-Computing effizient nutzen, werden, durch die steigende Rechenkapazität und dem Fortschritt in der Modellierung, benötigt.

Diese Arbeit konzentriert sich auf die mathematische Modellierung und numerische Simulation, um die Strömungsdynamik und die Verbrennungsprozesse innerhalb eines Verbrennungsmotors zu untersuchen. Ziel ist es, einfache, aber effiziente Ansätze zu entwickeln, um eine Reihe von Themen zu bewältigen, die für die Motorsimulationen wichtig sind. Diese Dissertation schlägt einen neuen Ansatz für die Motorsimulationen auf der Grundlage der Immersed Boundary Methode, Lagrange-Partikel und Grobstruktursimulationen (LES) vor. In dieser Arbeit wurde ein LES-Lösungsansatz für die vollständigen kompressiblen Navier-Stokes-Gleichung implementiert, um die Strömungsdynamik innerhalb eines Verbrennungsmotors zu simulieren. Zur modelierung der Interaktion zwischen Trägheitskräften des Fluides und den bewegenden Grenzen, wurde ein partikelbasiertes Verfahren entwickelt und in den Löser des kompressiblen Fluides implementiert. Durch die Anwendung dieser Methode auf Motorsimulationen, wird die Bewegung der Ventile und des Kolbens numerisch von den Lagrange-Partikeln beschrieben. Diese netzfreie Technik für die Gebietsrand-Darstellung ist einfach zu parallelisieren und eignet sich für Einsatz auf Hochleistungs-Rechnern. Die stationären Teile des Motors werden von einer rechnerisch effizienten Immersed Boundary Methode beschrieben, bei der die Netzerzeugung auf einem äquidistanten kartesischen Gitter schnell und einfach ist und die beteiligten numerischen Operationen durch Vektorisierung effizient durchgeführt werden können. Bei dem vorgeschlagenen Ansatz wurde die turbulente Verbrennung innerhalb des Brennraums durch das Flame-Surface-Density Model modelliert, wobei ein Einschritt-Mechanismus für die Wärmefreisetzung verwendet wurde.

Um das Verfahren zu demonstrieren, werden in dieser Arbeit LES-Ergebnisse für verschiedene Testfälle präsentiert z.B. die Strömung um einen Zylinder, in Kanälen, sowie Motorsimulationen aktueller Motorgeometrien mit und ohne Verbrennung. Die Ergebnisse demonstrieren die Eignung und die Zuverlässigkeit des vorgeschlagenen Verfahrens für präzise numerische Vorhersagen verschiedener Probleme in der numerischen Strömungssimulation, da gute Übereinstimmung zwischen den numerischen Ergebnissen

und den Experimenten erzielt wurde. Außerdem wird die Eignung der gewählten Vorgehensweise für Einsatz auf Hochleistungs-Rechnern durch massiv parallele Simulationen eines Verbrennungsmotors auf verschiedenen Supercomputern präsentiert.

Acknowledgements

This PhD work has been conducted under the supervision of Prof. Dr. Andreas Kempf at the Chair of Fluid Dynamics, University of Duisburg-Essen, Germany. I am indebted to Prof. Kempf for his guidance and encouragement. I would not have been able to complete my work without his continuous support.

I would like to thank Prof. Dr. Sebastian Kaiser and Dr. Irenäus Wlokas for many valuable discussions and suggestions which have led to significant improvement on my work.

I am fortunate to be a member of a team with many wonderful colleagues: Nejra Šikalo is often the first to read my writing, her comments and suggestions are invaluable to my work; I always enjoy discussing with Peter Janas to learn more from his insight into the field of internal combustion engine; Andreas Rittler has always brought critical questions into my work, his intuitive way of thinking amazed me; I have learned a lot from working with Fabian Proch who is extremely knowledgeable in many different subjects; Lei Deng has helped me a lot at work and his positive attitude in any circumstance is inspirational; Miriam Rabaçal is always a thoughtful colleague with remarkable insight and humor.

My sincere thanks also go to Syliva Helwig and Olaf Hasemann for their unfailing support and assistance during my time working at the Chair of Fluid Dynamics.

With a special mention to Claudia Weise, Khadijeh Mohri, Patrick Wollny, Eray Inanc, Johannes Sellmann, Martin Rieth, Timo Lipkowicz, Pascal Gruhlke, Vahid Sharifi, Hossein Janbazi, Luis Cifuentes, Stefan Wysocki and Terence Ma in general. I will miss our joyful and creative environment and I will never forget our Christmas celebrations.

Finally, I'm grateful to my parents, my sister and Hoai Le - my fiancée who have supported me along the way.

Duisburg, March 2017 Thuong Nguyen

Contents

List of Figures

List of Tables

Abbreviations

ATF	**A**rtificial **T**hickening **F**lame
BDC	**B**ottom **D**ead **C**enter
CMC	**C**onditional **M**oment **C**losure
DNS	**D**irect **N**umerial **S**imulation
EVO	**E**xhaust **V**alve **O**pening
EVC	**E**xhaust **V**alve **C**losing
FGM	**F**lame **G**enerated **M**anifold
FSD	**F**lame **S**urface **D**ensity
IBM	**I**mmersed **B**oundary **M**ethod
ICE	**I**nternal **C**ombustion **E**ngine
ILDM	**I**ntrinsic **L**ow **D**imensional **M**anifold
ISAT	**I**n **S**itu **A**daptive **T**abulation
IVO	**I**ntake **V**alve **O**pening
IVC	**I**ntake **V**alve **C**losing
LES	**L**arge **E**ddy **S**imulation
NSCBC	**N**avier-**S**tokes **C**haracteristic **B**oundary **C**onditions
RANS	**R**eynolds **A**veraged **N**avier **S**tokes
RPM	**R**evolutions **P**er **M**inute
SOC	**S**tart **O**f **C**ombustion
TDC	**T**op **D**ead **C**enter

Symbols

Latin symbols

A_{fj}	pre-exponential constant	——
c	progress variable	——
c_L	parameter in non-dimensional function f_L	——
c_v	specific heat capacity of the multicomponent mixture at a constant volume	$\mathrm{J/kg \cdot K}$
c_p	specific heat capacity of the multicomponent mixture at a constant pressure	$\mathrm{J/kg \cdot K}$
c_{pk}	specific heat capacity at a constant pressure	$\mathrm{J/kg \cdot K}$
c_{vk}	specific heat capacity at a constant volume	$\mathrm{J/kg \cdot K}$
c_η	parameter in non-dimensional function f_η	——
C	universal Kolmogorov constant	——
C_p	pressure coefficient	——
C_d	drag coefficient	——
C_s	model constant in Smagorinsky model	——
C_σ	model constant in Sigma model	——
Da_l	turbulent Damköhler number	——
D_k	diffusion coefficient of a species k	——
D_σ	differential operator in Sigma model	——
e	total energy	J
e_s	sensible internal energy	J
f_L	non-dimensional function	——
f_η	non-dimensional function	——
$\overline{f}(x,t)$	filtered quantity of $f(x,t)$	——
$f'(x,t)$	unresolved part of $f(x,t)$	——

$\tilde{f}(x,t)$	resolved quantity of $f(x,t)$ using mass-weighted Favre-filter	——
E_j	activation energy in Arrhenius expression	J/mol
$E(\kappa)$	energy of eddies size l and wavenumber κ	$\mathrm{m}^2/\mathrm{s}^2$
$F_{f,C}$	convective fluxes of Φ across surface f	——
$F_{f,D}$	diffusive fluxes of Φ across surface f	——
$G(r,x)$	filter function	——
$\boldsymbol{\mathcal{G}}$	velocity gradient tensor	——
$\Delta h^0_{f,k}$	mass enthalpy of formation for species k	J/kg
H_k	enthalpy of species k	$\mathrm{J/kg \cdot K}$
$\boldsymbol{I}_1$	invariant of velocity gradient tensor $\boldsymbol{G}$	——
$\boldsymbol{I}_2$	invariant of velocity gradient tensor $\boldsymbol{G}$	——
$\boldsymbol{I}_3$	invariant of velocity gradient tensor $\boldsymbol{G}$	——
k	kinetic energy	$\mathrm{m}^2/\mathrm{s}^2$
k_{sgs}	subgrid kinetic energy	$\mathrm{m}^2/\mathrm{s}^2$
Ka	Karlovitz number	——
K_{fj}	forward rate of reaction j	——
K_{rj}	reverse rate of reaction j	——
K_Σ	model constant in a FSD model proposed by Borger et al.	——
l_0	characteristic length of largest eddies	m
l_i	eigenvectors	——
L	characteristic length of the flow	m
L_i	characteristic waves in NSCBC	——
$\mathcal{L}_{ij}$	Germano identity	——
$\mathcal{M}_k$	symbol for species k	——
p	pressure	$\mathrm{N/m}^2$
p_k	pressure of a species k	$\mathrm{N/m}^2$
p^0	parameter in non-dimensional function f_L	——
p_∞	pressure at the far field	$\mathrm{N/m}^2$
Pr_t	Prandtl number	——
q_j	reaction rate variable of the reaction j	——
Re	Reynolds number	——
Re_{crit}	critical value of Reynolds number	——
Re_η	Reynolds number of the Kolmogorov length scales	——

Re_λ	Reynolds number of the Taylor length scales	——
Re_0	Reynolds number of largest eddies	——
Re_t	turbulent Reynolds number	——
s_{ij}	strain rate tensor	$\mathrm{m/s^2}$
s_L	laminar flame speed	m/s
S_k	entropy of species k	$\mathrm{J/kg \cdot K}$
S_Φ	source term of quantity Φ	——
$\overline{S}_{ij}$	resolved shear stress	$\mathrm{m/s^2}$
Sc_t	turbulent Schmidt number	——
T	temperature	K
T_0	standard-state temperature	K
T_u	unburned temperature	K
u_i	velocity component	m/s
u_0	characteristic velocity of largest eddies	m/s
u'	turbulence intensity	m/s
u_η	characteristic velocity of the Kolmogorov length scales	m/s
u_τ	friction velocity	m/s
U_∞	velocity at the far field	m/s
ΔV	a control volume	$\mathrm{m^3}$
$\partial\Delta V, A$	surface of a control volume	$\mathrm{m^2}$
W_k	atomic weight of species k	——
X_k	mole fraction of the component k	——
$[X_k]$	mole concentration of component k	$\mathrm{mol/m^3}$
Y_k	mass fraction of species k	——
Y_F	fuel mass fraction	——
Y_F^u	unburned state of fuel mass fraction	——
Y_F^b	burned state of fuel mass fraction	——

Greek symbols

α_i	angles computed from invariants of $\boldsymbol{G}$	——
β_j	temperature exponent	——
β	parameter in non-dimensional function f_η	——
γ	heat capacity ratio	——
Γ_k	Intermittent Turbulence Net Flame Stretch (ITNFS)	——

Γ_τ	wall shear stress	$\mathrm{m/s^2}$
δ_{ij}	Kronecker delta	——
δ_F	flame thickness	m
Δ	filter width (cell size)	m
ε	energy dissipation rate	$\mathrm{m^2/s^3}$
η	characteristic length of the Kolmogorov length scales	m
κ	wavenumber	——
λ	characteristic length of the Taylor length scales	m
λ	heat diffusion coefficient	——
μ	dynamic viscosity	$\mathrm{kg/m \cdot s}$
ν	kinematic viscosity	$\mathrm{m^2/s}$
ν_{kj}	stoichiometric coefficients of species k in reaction j	——
ν'_{kj}	stoichiometric coefficients of species k in reaction j	——
ν''_{kj}	stoichiometric coefficients of species k in reaction j	——
ν_t	sub-grid scale viscosity	$\mathrm{m^2/s}$
ρ	density	$\mathrm{kg/m^3}$
ρ_k	density of a species k	$\mathrm{kg/m^3}$
ρ_u	unburned density	$\mathrm{kg/m^3}$
σ_i	singular values of $\boldsymbol{G}$	——
σ_{ij}	Cauchy stress tensor	$\mathrm{N/m^2}$
Σ_{gen}	generalized flame surface density	$\mathrm{m^{-1}}$
τ_η	characteristic time of the Kolmogorov length scales	s
τ_λ	characteristic time of the Taylor length scales	s
τ_0	characteristic time of largest eddies	s
τ_c	chemical time scales	s
τ_{ij}	stress tensor	$\mathrm{N/m^2}$
Ξ	wrinkling factor	——
Φ	scalar quantity	——
$\Psi(r)$	limiter function	——
$\dot{\omega}_k$	source term	$\mathrm{kg/m^3 \cdot s}$
$\dot{\omega}_T$	heat release due to combustion	$\mathrm{J/kg \cdot s}$

Chapter 1

Introduction

The production rate of vehicles with combustion engines has increased by a factor of seven since 1950, and will continue to grow (see Fig. 1.1). It is estimated that the number of motorized vehicles will reach over two billion by 2020, and approximately half of them will be cars [1].

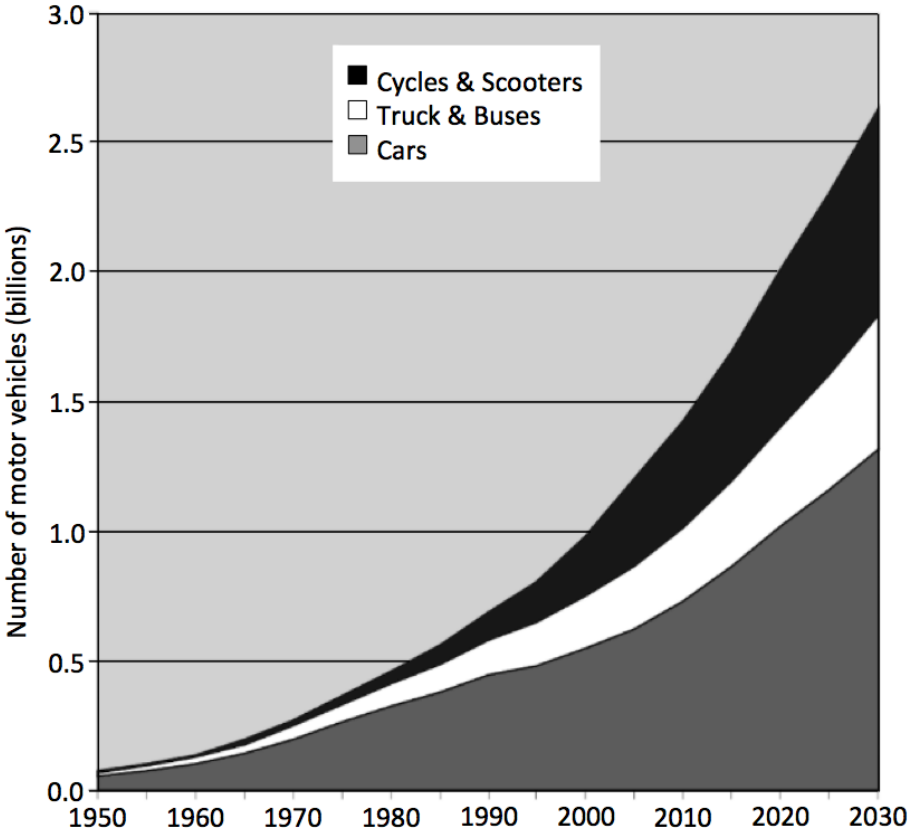

FIGURE 1.1: Historical and projected increase in the global motor vehicle (from Sperling et al. [1], p. 5).

Along with the rapid growth in the number of motorized vehicles comes the increase in the greenhouse gas emissions and air pollution, which are two major environmental concerns. Transportation contributes about 25% to the total global carbon dioxide (CO_2) emission [1]. Reducing the greenhouse gas emissions and the air pollution requires international commitments and world-wide efforts. Strict regulations on exhaust gas emissions have been introduced and implemented. For instance, the European emission standard for vehicles (EURO 1 to EURO 6) has become more strict in recent years, as illustrated in Fig. 1.2.

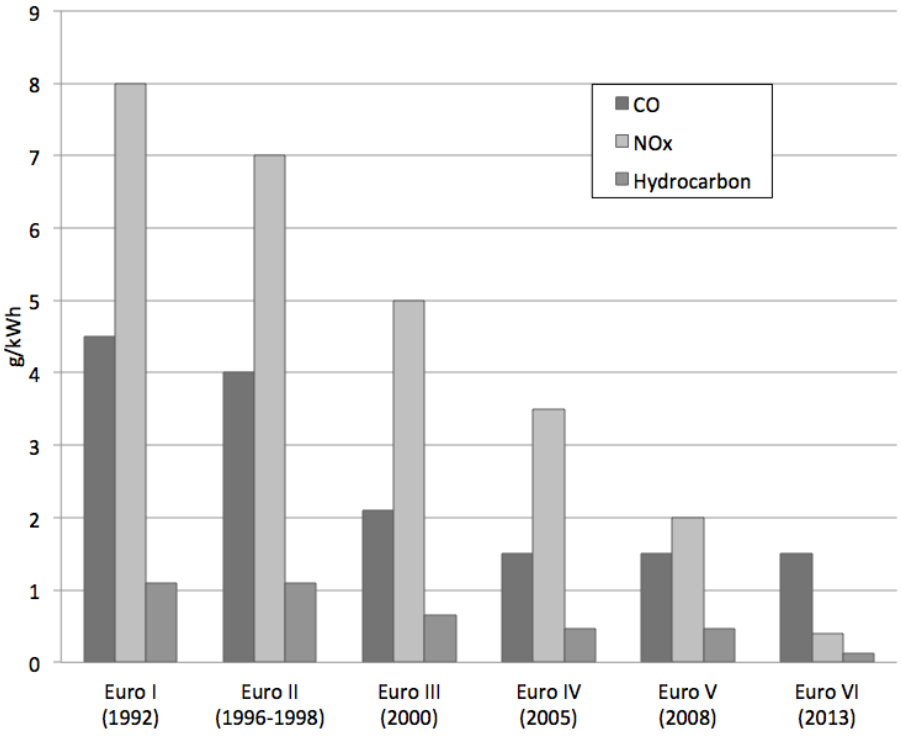

FIGURE 1.2: European emission standards for heavy duty Diesel engines [2].

To comply with these regulations, research efforts on clean Diesel- and gasoline engines play a central role in the development of new generation engines. Advancements in cleaner and more efficient combustion are as important as other measures such as alternative fuels, improved catalyst technologies or electronically controlled fuel injectors. Research on clean combustion requires a deep understanding of fluid dynamics and turbulent combustion, which occurs within the combustion chamber of an internal combustion engine (ICE). Besides the experimental-based methods, where the research and development of ICE strongly relies on laboratories with special equipment and highly-trained technicians, the rapid development of computer hardware and numerical methods additionally offer the simulation-based approach. Numerical simulations, in

contrast to the experiments, can provide more temporal and spatial insight of any physical quantity in a two- or three-dimensional field. Early studies on engine simulations by Bracco et al. [3], Boni et al. [4], Diwakar [5], Griffin et al. [6], Chapman [7], Takizawa et al. [8], Amsden et al. [9] and Khalighi et al. [10] have demonstrated the capability of numerical methods to capture the complicated two- and three dimensional flow inside an ICE. These investigations covered not only the flow dynamics but also the interaction between turbulence and chemistry in the reciprocating engines. The exponential growth in computational capacity allows the use of more expensive and complex numerical approaches such as Large Eddy Simulation (LES) or Direct Numerical Simulation (DNS), that can be efficiently used to simulate the flow and the combustion inside the ICE. The pioneering work on engine simulations, and the related issues using LES, can be found in the publications by Buttler et al. [11], Naitoh et al. [12], Han et al. [13], Celik et al. [14, 15] or Richard et al. [16].

The recent developments in combustion modelling have greatly improved the quality of ICE simulations, in which many complex physical phenomena can be well-captured. Modelling of chemistry in turbulent combustion has extensively progressed with efficient techniques to reduce and tabulate chemical schemes such as intrinsic low dimensional manifold (ILDM) [17, 18], flame prolongation of ILDM [19], flame generated manifold (FGM) [20] or in-situ adaptive tabulation (ISAT) [21, 22]. These techniques allow a full description of chemical reactions during the combustion process, where hundreds of species and thousands of reactions are involved. Together with these tabulated chemistry techniques, elegant approaches like G-equations [23, 24, 25, 26, 27], flame surface density (FSD) [28, 29, 30, 31, 32], artificial flame thickening (ATF) [33, 34, 35, 36, 37, 38] or conditional moment closure (CMC) [39, 40, 41, 42, 43, 44] are used to accurately predict flame propagation under the influence of turbulence.

Besides the main issues related to the in-cylinder flow and combustion [45, 46, 47, 48, 49, 50], the LES of ICE also tackles many complex issues such as NOx emissions [51, 52, 53, 54, 55, 56], knock [57, 58, 59, 60], cycle-to-cycle variations (CCV) [61, 62, 63, 64, 65, 66, 60] or flame-acoustics interactions [67, 68, 69, 70]. As the demands for low emissions and higher efficiency of the ICE keep increasing, the development of mathematical and numerical methodologies for the modelling of turbulent combustion are therefore of crucial interest. Finally, developing numerical tools for ICE simulations, that can solve

a wide range of complex industrial problems, will still be subject to ongoing studies for many years to come.

1.1 Motivation

Numerical modelling of an ICE is complicated and challenging in many different ways. First, mesh generation of the engine geometry with sophisticated features such as the cylinder head and the intake- and the exhaust manifolds is difficult, not to mention the moving boundaries like the piston and the valves. Second, the dynamics of the fluid is governed by the interaction between the moving boundaries and the fluid inertia in a confined domain. Third, the combustion process inside an ICE is not yet fully understood and is difficult to simulate accurately, since the flame propagation depends on the turbulence that is created and maintained throughout a long process, from the intake to compression. Finally, it should be stressed that the simulations of an ICE are often time-consuming and computationally expensive since the statistics requires a large number of engine cycles to capture not only the physics of the fluid itself, but the so-called cycle-to-cycle variations too.

Focusing on the mathematical modelling and the numerical simulations of ICE, the first aim of this study is to find simple but efficient numerical approaches to tackle the following issues: The mesh generation for engine geometries, the numerical description of the moving boundaries, the interaction between the fluid flow and the moving objects and the modelling of turbulent combustion. The second aim of this work is to validate the employed numerical approaches by comparing the obtained results with experimental data from literature and from the research group at the Technical University of Darmstadt.

To accomplish the aforementioned goals, this work proposes a new approach for engine simulation which is easy to implement, efficient for high-performance computing and simple for mesh generation. In the new approach, complicated body-fitted meshes are replaced by an equidistant Cartesian grid using the immersed boundary method (IBM). The moving boundary is tackled in a simple and flexible manner by using a particle-based method instead of complex dynamic mesh generation. To accurately simulate the flame propagation during the fire-stroke, a simple combustion model (flame surface density)

is employed with a single-step mechanism for the calculation of the heat release. The proposed approach is then validated with the simulated- and the experimental data.

1.2 Thesis outline

This thesis is structured as follows: Chapter 2 introduces the fundamentals of turbulence, chemistry and turbulent combustion. The LES of turbulence and turbulent combustion is explained in detail too. Chapter 3 presents the numerical approaches for handling a wide range of problems including the spatial and temporal discretization of a convection - diffusion problem, the modelling of turbulence and turbulent combustion, the numerical treatments of the moving boundary and the non-reflecting boundary in a compressible flow. In Chapter 4, the validation of the IBM and the Lagrangian particle approach for moving boundaries is carried out. Chapter 5 is structured according to the published SAE technical paper, where the numerical simulations of a four-stroke engine from the University of Duisburg-Essen were performed by two different numerical approaches: OpenFOAM and PsiPhi. The obtained results were compared against each other and against the experimental data provided by the research group at the University of Duisburg-Essen. Chapter 6 presents a multi-cycle engine simulation of the Darmstadt engine, which was performed for the motor- and the fired-cases. This chapter corresponds to the paper published in the journal of Flow, Turbulence and Combustion. Investigation of the numerical effects on the in-cylinder flow and the combustion process of an ICE is carried out in Chapter 7, corresponding to the paper submitted to Oil & Gas Science and Technology - Revue d'IFP Energies nouvelles. In this study, multi-cycle engine simulations were performed using different numerical schemes. Detailed analysis of the physical and the numerical quantities of the in-cylinder flow field is carried out. The conclusions and outlook are given in Chapter 8.

Chapter 2

Theoretical background

This chapter presents the basic mathematical models of the conservation laws. The governing equations for the fluid motion and some concepts of turbulence are introduced. In the context of engine simulations, the understanding of thermodynamics, chemical kinetics and turbulent combustion is essential, therefore, the relevant background information of these topics are also covered by this chapter. Finally, the turbulence and the combustion modelling in LES are discussed in detail.

2.1 Navier-Stokes equations

The conservation of mass, momentum and energy of fluid flow are described by the Navier-Stokes equations. By solving these equations, the physical quantities such as the velocity, temperature and pressure fields of any system can be obtained. The transport of the conserved quantities is represented with the following set of equations [71].

The continuity equation is given by:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \tag{2.1}$$

The momentum equation is expressed as:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \tag{2.2}$$

In equations 2.1 and 2.2, ρ, u_i, p and τ_{ij} denote the density, velocity, pressure and stress tensor, respectively.

The viscous stress tensor τ_{ij} of a Newtonian fluid is defined as follows:

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) \tag{2.3}$$

$$\text{and} \quad \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}, \tag{2.4}$$

where μ and δ_{ij} are the dynamic viscosity and Kronecker delta, respectively.

The species transport equation in reactive flows is shown below:

$$\frac{\partial \rho Y_k}{\partial t} + \frac{\partial \rho Y_k u_i}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\rho D_k \frac{\partial Y_k}{\partial x_i}\right) + \dot{\omega}_k \quad (k = 1, ..., n) \tag{2.5}$$

In equation 2.5, Y_k, $\dot{\omega}_k$ and D_k represent the mass fraction, source term and diffusion coefficient of species k. For reactions with n species, the sum of the mass fractions is unity $\sum_{k=1}^{n} Y_k = 1$ and the sum of the reaction source terms is null $\sum_{k=1}^{n} \dot{\omega}_k = 0$.

The transport equation of energy which $e = [0.5 u_k u_k + e_s]$ reads:

$$\frac{\partial \rho e}{\partial t} + \frac{\partial \rho e u_i}{\partial x_i} = \dot{\omega}_T + \frac{\partial}{\partial x_i}\left(\lambda \frac{\partial T}{\partial x_i}\right) + \frac{\partial}{\partial x_i}\left(u_j \sigma_{ij}\right) \tag{2.6}$$

In Eq. 2.6, $\dot{\omega}_T$, T, λ and σ_{ij} are the heat release due to combustion, the temperature, the heat diffusion coefficient and the Cauchy stress tensor, respectively. The Cauchy stress tensor σ_{ij} is defined as:

$$\sigma_{ij} = \tau_{ij} - p\delta_{ij} \tag{2.7}$$

The heat release due to combustion $\dot{\omega}_T$ is computed from the formation enthalpy $\Delta h^o_{f,k}$ and source term $\dot{\omega}_k$:

$$\dot{\omega}_T = -\sum_{k=1}^{N} \Delta h^o_{f,k} \dot{\omega}_k \tag{2.8}$$

The sensible internal energy e_s is determined according to:

$$e_s = \int_{T_0}^{T} C_v(T)\, dT \tag{2.9}$$

where C_v represents for the specific heat capacity at constant volume.

2.2 Thermochemical properties

The chemical state of the system can be determined by the laws of thermodynamics and reaction kinetics. Following thermal equation of state for an ideal gas, the relation between the partial pressure p_k, partial density ρ_k, atomic weight W_k of species k and temperature T of the mixture is expressed as [71]:

$$p_k = \rho_k \frac{R}{W_k} T, \quad (k = 1, ..., N) \tag{2.10}$$

The pressure p, the density ρ and the molecular mass W of the mixture are evaluated according to:

$$p = \sum_{k=1}^{N} p_k \ , \tag{2.11}$$

$$\rho = \sum_{k=1}^{N} \rho_k \ , \tag{2.12}$$

$$\frac{1}{W} = \sum_{k=1}^{N} \frac{Y_k}{W_k} \ . \tag{2.13}$$

In Eq. 2.10, $R = 8.314$ J/mol·K represents the ideal gas constant. The partial density of species k can directly be calculated from the mass fraction $\rho_k = \rho Y_k$.

In a reactive flow, the sensible internal energy of the mixture is computed from:

$$\begin{aligned} e &= \sum_{k=1}^{N} \left(\int_{T_0}^{T} C_{vk} dT - RT_0/W_k + \Delta h^0_{f,k} \right) Y_k \\ &= \int_{T_0}^{T} C_v dT - RT_0/W + \sum_{k=1}^{N} \Delta h^0_{f,k} Y_k \end{aligned} \tag{2.14}$$

The enthalpy of the mixture is calculated as:

$$h = \sum_{k=1}^{N} h_k Y_k = \sum_{k=1}^{N} \left(\int_{T_0}^{T} C_{pk} dT + \Delta h_{f,k}^0 \right) Y_k = \int_{T_0}^{T} C_p dT + \sum_{k=1}^{N} \Delta h_{f,k}^0 Y_k \ , \tag{2.15}$$

where $\Delta h_{f,k}^0$ denotes the specific enthalpy of formation of species k at the reference temperature $T_0 = 298.15$ K. The relation between the heat capacities of species k at constant volume C_{vk} and constant pressure C_{pk} is:

$$C_{pk} - C_{vk} = \frac{R}{W_k} \tag{2.16}$$

Specific heats in constant volume and constant pressure of the mixture are calculated as mass weighted according to:

$$\begin{aligned} C_v &= \sum_{k=1}^{N} C_{vk} Y_k, \\ C_p &= \sum_{k=1}^{N} C_{pk} Y_k \end{aligned} \tag{2.17}$$

The temperature dependence of thermochemical properties of a chemical substance can be described by the NASA polynomials [72]:

$$\begin{aligned} \frac{C_{pk}}{R} &= a_{1k} + a_{2k} T_k + a_{3k} T_k^2 + a_{4k} T_k^3 + a_{5k} T_k^4 \ , \\ \frac{H_k}{R T_k} &= a_{1k} + \frac{a_{2k}}{2} T_k + \frac{a_{3k}}{3} T_k^2 + \frac{a_{4k}}{4} T_k^3 + \frac{a_{5k}}{5} T_k^4 + \frac{a_{6k}}{T_k} \ , \\ \frac{S_k}{R} &= a_{1k} ln T_k + a_{2k} T_k + \frac{a_{3k}}{2} T_k^2 + \frac{a_{4k}}{3} T_k^3 + \frac{a_{5k}}{4} T_k^4 + a_{7k} \end{aligned} \tag{2.18}$$

The numerical coefficients a_{ik} are provided from the thermochemical database. H_k and S_k represent the enthalpy and the entropy of species k.

2.3 Chemical kinetics

This section follows closely the books of Poinsot et al. [71] and Kee et al. [73]. A chemical reaction is a process that transforms a set of substances (reactants) to another (products). For example, the global reaction of iso-octane C_8H_{18} and oxidizer O_2 is

described by the following equation:

$$C_8H_{18} + \frac{25}{2}O_2 \rightarrow 8CO_2 + 9H_2O$$

Here the chemical process converts the reactants iso-octane C_8H_{18} and oxidizer O_2 to the products carbon dioxide CO_2 and water H_2O. In a chemical reaction, the atoms (C, H, O) are conserved. In fact, the global reaction is the result of many elementary reactions, for example:

$$\begin{aligned} HO_2 + OH &\rightleftharpoons H_2O + O_2 \\ HO_2 + H &\rightleftharpoons OH + OH \\ H + O_2(+M) &\rightleftharpoons HO_2(+M) \\ H + O_2 &\rightleftharpoons O + OH \\ CO + OH &\rightleftharpoons CO_2 + H \\ HCO + H &\rightleftharpoons CO + H_2 \\ &\vdots \end{aligned}$$

Considering the elementary reactions of a multicomponent mixture (N species), the interaction between the species (M reactions) can be written in the general form [71]:

$$\sum_{k=1}^{N} \nu'_{kj} \mathcal{M}_k \rightleftharpoons \sum_{k=1}^{N} \nu''_{kj} \mathcal{M}_k, \quad (j = 1, ..., M) \ , \tag{2.19}$$

where $\mathcal{M}_k$ denotes a symbol for species k, ν'_{kj} and ν''_{kj} represent the stoichiometric coefficients of species k in reaction j.

Since the mass is conserved, this leads to:

$$\sum_{k=1}^{N} \nu'_{kj} W_k = \sum_{k=1}^{N} \nu''_{kj} W_k, \quad (j = 1, ..., M) \tag{2.20}$$

The source term $\dot{\omega}_k$ of species k in Eq. 2.21 is obtained by summing up the reaction rates of the species in a system and is outlined below:

$$\begin{aligned} \dot{\omega}_k &= W_k \sum_{j=1}^{M} \nu_{kj} q_j \ , \\ \text{with} \ \ \nu_{kj} &= \nu''_{kj} - \nu'_{kj} \end{aligned} \tag{2.21}$$

The reaction rate variable q_j of the reaction j is calculated as the difference between the forward and backward rates:

$$q_j = K_{fj} \prod_{k=1}^{N} [X_k]^{\nu'_{kj}} - K_{rj} \prod_{k=1}^{N} [X_k]^{\nu''_{kj}} \tag{2.22}$$

In Eq. 2.22, $[X_k]$ is the mole concentration of species k and can be expressed as $[X_k] = \rho Y_k / W_k$. K_{fj} and K_{rj} represent the forward and backward rates of reaction j which can be estimated from the empirical Arrhenius law:

$$K_{fj} = A_{fj} T^{\beta j} \exp\left(- \frac{T_{aj}}{T} \right) \tag{2.23}$$

In Eq. 2.23, A_{fj}, β_j and E_j represent the pre-exponential constant, the temperature exponent and the activation energy, respectively. For further information, readers are referred to the books of Poinsot et al. [71] or Kee et al. [73].

2.4 Turbulence

The following section closely follows the book of Pope [74]. Flows can be classified into laminar and turbulent regimes, based on the Reynolds number, which is the ratio of the inertia forces to the viscous forces:

$$\mathrm{Re} = \frac{uL}{\nu} \tag{2.24}$$

In Eq. 2.24, u, L and ν are the characteristic velocity, the characteristic length of the flow and the kinematic viscosity, respectively. When Re of a flow exceeds a critical value Re_{crit} ($\mathrm{Re} > \mathrm{Re}_{crit}$), the flow behaves randomly and chaotically, which falls into the turbulent regime. Below the critical value ($\mathrm{Re} < \mathrm{Re}_{crit}$), a fluid flows smoothly and orderly in separate parallel layers, which is known as laminar flow.

Turbulence is a common physical phenomenon in nature: the flows in the rivers, the currents in the deep oceans, the air streams in the earth's atmosphere, the gas flow inside an internal combustion engine are turbulent. Turbulence happens at multiple scales in time and space, and it has been well summarised by Richardson [75]:

'Big whirls have little whirls that feed on their velocity.
And little whirls have littler whirls, and so on to viscosity.'

From the description of turbulence by Richardson, some important insights of the turbulent eddies are provided.

- Turbulent flows contain eddies of different sizes.
- The large eddies become unstable and break up into smaller eddies.
- The energy is transferred down from the large eddies to smaller eddies until the smallest length scale is reached where the energy is dissipated by the viscosity of the fluid.

2.4.1 Length scales of turbulent flow

In order to follow the mathematical perspectives of turbulent flows, it is important to introduce the Reynolds averaging as a process of averaging a variable, Reynolds averaging can also be done for samples or phases, in time. Considering a dependent variable Φ that varies in time, Φ can be decomposed into an average part $\overline{\Phi}$ and a fluctuation part Φ', which is known as Reynolds decomposition:

$$\Phi' \equiv \Phi - \overline{\Phi} \tag{2.25}$$

The Reynold averaging is introduced as follows:

$$\overline{\Phi} \equiv \lim_{T\to\infty} \frac{1}{T} \int_T \Phi(t)dt \tag{2.26}$$

The time period T in Eq. 2.26 should be long enough to average out the fluctuation in Φ. By applying the Reynolds decomposition to the velocity field u_i and the pressure in the momentum equation, and subsequently Reynolds averaging to the decomposed equations yields the Reynolds stress tensor $\overline{u_i' u_j'}$. The Reynolds stress tensor, results from the convection term of the momentum equation, remains unclosed. Commonly it may be assumed that the turbulence on the small scales is isotropic where the turbulent quantities are directionally-independent. By employing the Boussinesq hypothesis [76],

the Reynolds stress tensor may be approximated in the following way:

$$-\rho\overline{u_i'u_j'} = \mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) - \frac{2}{3}\rho k\delta_{ij} \tag{2.27}$$

In Eq. 2.27, μ_t is the turbulent viscosity and k represents the turbulent kinetic energy, which is defined as:

$$k = \frac{1}{2}\left(\overline{u_i'u_i'}\right) = \frac{1}{2}\left(\overline{u_1'^2} + \overline{u_2'^2} + \overline{u_3'^2}\right) \tag{2.28}$$

To gain further understanding of turbulence, important aspects of turbulent eddies are considered: (1) turbulent eddies are characterized by the size l, characteristic velocity $u(l)$ and timescale $t(l) = l/u(l)$; (2) the length scale l_0 of eddies in the largest size range is equivalent to the flow length scale L, where the characteristic velocity $u(l_0)$ of these largest eddies is equivalent to the flow velocity U. As a matter of fact, the Reynolds number $\mathrm{Re}_0 = u_0 l_0/\nu$ of these largest eddies is comparable to the Reynolds number $\mathrm{Re} = UL/\nu$ of the fluid flow.

Eddies can be classified into many length scales, amongst them the integral length scales, the Taylor microscales, and the Kolmogorov length scales.

Integral length scales

An integral length scale is a quantitative measure of large scale structures, which can be considered as the correlation between velocity components at two separate spatial locations in the flow [77]. The integral length scales are the largest scales in the energy spectrum which contain most of the energy. In these scales, the fluid flow is highly anisotropic and the energy is transferred between the eddies themselves or the eddies and the mean flow. Since the characteristic velocity $u_0(l_0)$ is in the order of turbulence intensity $\sqrt{2k/3}$, the relation between the integral length scale l_0 [m], the kinetic energy k [$\mathrm{m}^2/\mathrm{s}^2$] and the energy dissipation rate ε [$\mathrm{m}^2/\mathrm{s}^3$] is:

$$l_0 \propto \frac{k^{3/2}}{\epsilon} \tag{2.29}$$

The turbulent Reynolds number corresponding to these large eddies is

$$\mathrm{Re}_t = \frac{u_0 l_0}{\nu} = \frac{k^{1/2} l_0}{\nu} = \frac{k^2}{\varepsilon\nu} \tag{2.30}$$

Kolmogorov length scales

Kolmogorov length scales are the smallest scales in the energy spectrum where the energy is dissipated by the viscosity of the fluid. The turbulence in the Kolmogorov length scales is considered to be locally isotropic, homogeneous and high in frequency. At these smallest scales, the fluid motions are statistically independent from the mean flow field and the boundary conditions. These scales can be determined by the energy dissipation rate ε and the kinematic viscosity ν. The Kolmogorov length scales are characterized by the characteristic length η:

$$\eta = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4} , \tag{2.31}$$

the characteristic velocity u_η:

$$u_\eta = \left(\nu\varepsilon\right)^{1/4} , \tag{2.32}$$

the characteristic time τ_η:

$$\tau_\eta = \left(\frac{\nu}{\varepsilon}\right)^{1/2} , \tag{2.33}$$

and the Reynolds number Re_η:

$$\mathrm{Re}_\eta = \frac{\eta u_\eta}{\nu} = 1 \tag{2.34}$$

Taylor microscales

Between the integral length scales l_0 and the smallest Kolmogorov length scales η are the Taylor microscales λ, where the energy is passed down from the larger to the smaller eddies, which is referred to as the energy cascade. The energy dissipation rate in Taylor microscales is calculated as:

$$\varepsilon = 2\nu\overline{s_{ij}s_{ij}} , \tag{2.35}$$

where the strain rate tensor s_{ij} is computed from the fluctuating component of the velocity:

$$s_{ij} = \frac{1}{2}\left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i}\right) . \tag{2.36}$$

In the case of isotropic turbulence ($u_1' = u_2' = u_3'$), the energy dissipation rate becomes

$$\varepsilon = 15\nu\overline{\left(\frac{\partial u_1'}{\partial x_1}\right)^2} \,, \tag{2.37}$$

considering

$$\overline{\left(\frac{\partial u_1'}{\partial x_1}\right)^2} \equiv \frac{\overline{u_1'^2}}{\lambda^2} = \frac{u_1'^2}{\lambda^2} \,. \tag{2.38}$$

The energy dissipation rate becomes $\varepsilon = 15\nu u_1'^2/\lambda^2$ and the Taylor microscales λ can be estimated:

$$\lambda \approx \left(\frac{10\nu k}{\varepsilon}\right)^{1/2} \,, \tag{2.39}$$

where the turbulent kinetic energy $k = (1/2)\left(\overline{u_1'^2} + \overline{u_2'^2} + \overline{u_3'^2}\right) = (3/2)u_1'^2$.

The Taylor micorscales describe the transition of the eddies from the inertial sub-range to the dissipation range. The corresponding Reynolds number and the characteristic time scale of the Taylor microscales are given as:

$$\mathrm{Re}_\lambda = \frac{u'\lambda}{\nu} \,, \tag{2.40}$$

$$\tau_\lambda = \frac{\lambda}{u'} \,. \tag{2.41}$$

The relation between Taylor microscales, integral length scales and Kolmogorov scales yields:

$$\frac{\lambda}{l_0} = \sqrt{10} Re_L^{-1/2} \,, \tag{2.42}$$

$$\frac{\lambda}{\eta} = \sqrt{10} Re_L^{-1/4} \,, \tag{2.43}$$

$$\lambda = \sqrt{10}\eta^{2/3} l_0^{1/3} \,. \tag{2.44}$$

2.4.2 Energy spectrum

Considering $E(\kappa)$ is the energy, which is contained in eddies of size l and κ is the wavenumber $\kappa = 2\pi/l$, the turbulent kinetic energy k over all wavenumbers is defined

as:

$$k = \int_0^\infty E(\kappa) d\kappa \ . \tag{2.45}$$

According to Eq. 2.45, the energy contained in eddies in a range of wavenumbers κ_A and κ_B is simply evaluated by

$$k_{(\kappa_A,\kappa_B)} = \int_{\kappa_A}^{\kappa_B} E(\kappa) d\kappa \ . \tag{2.46}$$

Using dimensional analysis, the full energy spectrum (see Fig. 2.1) is obtained from [74]:

$$E(\kappa) = C\varepsilon^{2/3}\kappa^{-5/3} f_L f_\eta \ , \tag{2.47}$$

where the specified non-dimensional functions f_L and f_η are described by Pope [74] as follows:

$$f_L = \left(\frac{\kappa l_0}{[(\kappa l_0)^2 + c_L]^{1/2}} \right)^{p_0+5/3} \tag{2.48}$$

$$f_\eta = \exp\Big(- \beta\big([(\kappa\eta)^4 + c_\eta^4]^{1/4} - c_\eta\big) \Big) \ . \tag{2.49}$$

In Eqs. 2.47-2.49, the universal Kolmogorov constant $C = 1.5$ was obtained from experiments and the values of other parameters are given as $c_L \approx 6.78$, $c_\eta \approx 0.40$, $p_0 = 2$ and $\beta = 5.2$ [74].

In the inertial sub-range, where both f_L and f_η become unity, the energy spectrum is simplified as

$$E(\kappa) = C\varepsilon^{2/3}\kappa^{-5/3} \tag{2.50}$$

The full energy spectrum of a turbulent flow is illustrated in Fig. 2.1. It is determined that 80% of the turbulent kinetic energy is contained in the length scales l between $l_0/6$ and $6l_0$ ($l_0/6 < l < 6l_0$). Readers are referred to the work of Pope [74] for further details.

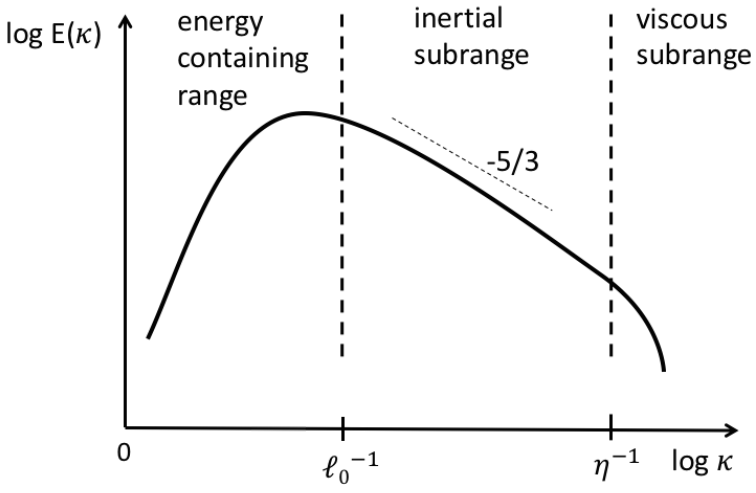

FIGURE 2.1: Energy cascade of Kolmogorov spectrum [74].

2.5 Turbulent combustion

This section focuses on turbulent combustion, which results from two-way interaction between turbulence and chemistry. The turbulence has a significant influence on combustion, such as altering the turbulent flame speed and flame structure. In some cases intense turbulence may quench or blow off the flame. Combustion, on the other hand, increases the gas temperature through the heat release, which results in a strong acceleration of the fluid flow due to dilatation [71].

The utilisation of the turbulent combustion can be found in many industrial applications such as spark-ignition engines, diesel engines, gas turbines, jet engines, rocket engines or in furnaces. In general, turbulent combustion can be practically classified in two groups: premixed and non-premixed combustion.

In the premixed combustion, the mixing between fuel and air happens relatively long before combustion occurs. For instance, in spark-ignition engines, the flame-kernel is initiated by spark discharge after air and fuel are mixed and compressed. Under the influence of turbulence, the small flame kernel will grow and expand rapidly into the unburned gases with the turbulent burning velocity.

On the other hand, the non-premixed combustion occurs instantly during the mixing process between fuel and air. Diesel engines are typical examples for this type of combustion. Instead of using spark-discharge to start the combustion, auto-ignition is the

main mechanism to ignite and rapidly burn the partially premixed gas of evaporating fuel droplets and the hot compressed air under very high in-cylinder pressure.

The chemistry and the turbulent time scales are important factors that determine the characteristics of the turbulence-chemistry interaction. The turbulent time scales depend on the dynamics of the fluid flow, in which the intense turbulence leads to a shorter turbulent time scales. In a reacting flow, the chemical time scales are strongly influenced by the temperature of the gas flow. A system, which comprises a reactive flow with short turbulent and long chemical time scales is referred to as slow chemistry, and vice versa. Reacting flows with fast chemistry are employed in almost all combustion applications since maintaining stable and efficient combustion requires high temperature of the mixture and therefore short chemical time scales [78].

In the scope of this work, premixed turbulent combustion is the main focus. Therefore, the fundamental knowledge of this topic is introduced. This section closely follows the books from Poinsot et al. [71] or Peters [79].

In the context of the premixed turbulent combustion, the flame thickness δ_F is determined from the laminar flame speed s_L and the diffusivity D:

$$\delta_F = \frac{D}{s_L} \tag{2.51}$$

The turbulent Reynolds number Re_t can be defined using the velocity fluctuation u', the integral length scales l_0, the laminar flame speed s_L and the flame thickness δ_F [71]:

$$\mathrm{Re}_t = \frac{u'}{s_L}\frac{l_0}{\delta_F} \tag{2.52}$$

The turbulent Damköhler number is defined as the ratio of the integral time scale τ_0 to the chemical time scale τ_c:

$$Da_l = \frac{\tau_0}{\tau_c} = \frac{l_0/u'}{\delta_F/s_L} \,. \tag{2.53}$$

The Damköhler number $Da_l \gg 1$ implies that the chemical times are shorter than the turbulent times, in which the inner flame structures are not influenced by the turbulence. Therefore, the mean burning rate can be estimated as a product of the burning rate of a laminar flame and the over-all flame surface [71]. In the case of the Damköhler number

$Da_l \ll 1$, turbulence has a strong influence on the combustion processes since the reactants and the products are mixed by fluid motions, which may decrease the burning rate. The turbulent Karlovitz number is defined as the ratio of the chemical time scales τ_c to the Kolmogorov time scales τ_η

$$Ka = Da_\eta^{-1} = \frac{\tau_c}{\tau_\eta} = \frac{\delta_F^2}{\eta^2} = \frac{u_\eta^2}{s_L^2} \ . \tag{2.54}$$

To derive the relation between u'/s_L and l_0/δ_F, the equation 2.52 is rewritten as:

$$\frac{u'}{s_L} = \mathrm{Re}_t \frac{\delta_F}{l_0} = Ka^{2/3} \left(\frac{l_0}{\delta_F}\right)^{1/3} \ . \tag{2.55}$$

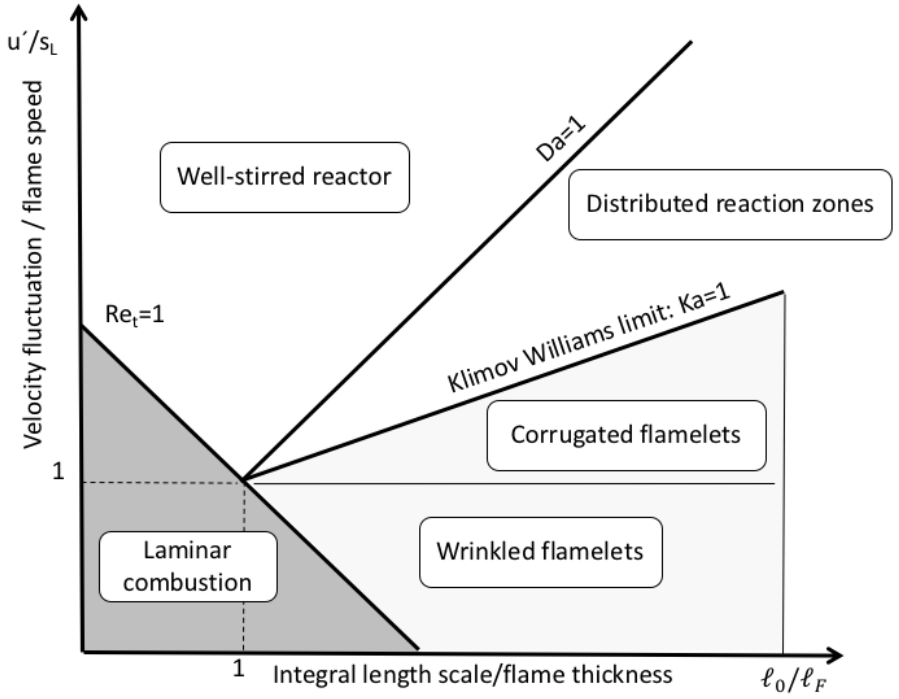

FIGURE 2.2: Classical diagram for premixed combustion regimes (log-log scale), as adapted from Peter [80].

Figure 2.2 shows different premixed combustion regimes which are distinguished using the ratios u'/s_L and l_F/l_0. From Fig. 2.2, few observations can be made:

- $Ka < 1$ corresponds to $\tau_c < \tau_\eta$ and $\delta_F < \eta$ (see Eq. 2.54), the wrinkled flame dominates the combustion regime: The chemical time scale τ_c is shorter than the turbulent time scale τ_η and the flame thickness δ_F is smaller than the Kolmogorov length scale η. This combustion regime is characterized by a thin flame front and an inner structure close to a laminar flame. In case of $u' < s_L$, we have a "wrinkled

flamelet regime" where the turbulence is too low to be able to wrinkle the flame. When $u' > s_L$, the influence of the turbulence on the flame is strong, in which the flame front is wrinkled and this leads to the interactions of the flame front to form pockets of fresh or burnt gases. In this case, we have a "corrugated flamelet regime".

- The region where $Ka > 1$ and $Da_l > 1$ corresponds to $\tau_\eta < \tau_c < \tau_0$ and $\eta < \delta_F$. In this case, the inner structure of the flame is modified by turbulence since the Kolmogorov length scales η are smaller than the flame thickness δ_F. Therefore, "distributed reaction zones" or "thickened flame regime" is dominant.

- $Da_l < 1$ implies that the turbulent time scales are shorter than the chemical time scales $\tau_0 < \tau_c$. In this regime, the reaction rate depends only on the chemistry time scales τ_c since the turbulent mixing is fast. This is called a "well-stirred reactor" regime.

Although it is quite helpful to provide some ideas about different combustion regimes, the analysis based on the classical diagram (see Fig. 2.2) for the turbulent premixed combustion fails in many cases due to several reasons: (1) the assumption of an homogeneous and isotropic frozen turbulence is unrealistic since the turbulence is strongly influenced by the heat release; (2) determination of criteria and combustion regime limits in the classical diagram is solely based on order of magnitude estimations rather than the precise derivations, which is obviously inaccurate. Further explanations can be found in the books of Poinsot et al. [71] or Peters [81].

2.6 Turbulence modelling with LES

This section closely follows the books from Poinsot et al. [71] and Pope [74]. Solving Navier-Stokes equations in a wide range of spatial and time scales for realistic cases is not feasible due to the limits of computational resources. Among the existing approaches, direct numerical simulation (DNS) is only applicable in certain limited situations with reasonable domain sizes and moderate Reynolds numbers. This is clearly not suitable for most of the industrial applications, where the size of the computational domain and the Reynolds number are large. Computational fluid dynamics in many industrial

applications requires practical approaches, where certain flow and time scales must be effectively resolved under the limited computational resources. Applying a low-pass filter to the Navier-Stokes equations, large-eddy simulations (LES) resolve only the flow scales that are larger than a chosen filter size (filter size $\Delta \gg \eta$), while modelling the smaller scales. In this section, the LES filter and the filtered form of the Navier-Stokes equations are introduced. Existing approaches for the modeling of unresolved stresses are presented.

2.6.1 LES filters

In order to remove the small scales from the simulation, a LES filter can be generally employed to perform the filtering operation on the spatial or temporal field. Using a filter function $G(x, r)$, the general filtered form of any quantity $f(x, t)$ is defined as [82]:

$$\overline{f}(x,t) = \int G(r,x) f(r,t) dr \ , \tag{2.56}$$

where the integration of the filtered function $G(r, x)$ over the entire domain satisfies:

$$\int G(r,x) dr = 1 \tag{2.57}$$

The definition of residuals field reads:

$$f'(x,t) \equiv f(x,t) - \overline{f}(x,t) \tag{2.58}$$

The decomposition of the velocity field follows:

$$\mathbf{U}(x,t) = \overline{\mathbf{U}}(x,t) + \mathbf{u}'(x,t) \ . \tag{2.59}$$

Unlike the Reynolds decomposition, in the LES filter operation, the filtered component $\overline{\mathbf{u}'}(x,t)$ is not equal to zero $\overline{\mathbf{u}'}(x,t) \neq 0$. Figure 2.3 illustrates some common LES filters in physical or spectral spaces. The formulations of these filters are presented in the following. More information on the LES filtering can be found in the book of Pope [74].

Box filter in physical space

The filter function for averaging over a cubic box has the form of

$$G(r) = G(x_1, x_2, x_3) = \begin{cases} 1/\Delta^3 & \text{if } |x_i| \leq \Delta/2 \text{ i=1,2,3} \\ 0 & \text{otherwise} \end{cases} , \tag{2.60}$$

where x_1, x_2 and x_3 are the spatial coordinates and Δ denotes the filter width.

Gaussian filter in physical space

The filter is simply a normalized Gaussian function and has the form of

$$G(r) = G(x_1, x_2, x_3) = \big(\frac{6}{\pi\Delta^2}\big)^{1/2} \exp\big[- \frac{6}{\Delta^2}(x_1^2 + x_2^2 + x_3^2)\big] \tag{2.61}$$

Cut-off filter in spectral space

Depending on the characteristic cut-off wavenumber, the cut-off filter in spectral space is similar to the box filter:

$$G(k) = \begin{cases} 1 & \text{if } |k| \leq k_c = \pi/\Delta \\ 0 & \text{otherwise} \end{cases} , \tag{2.62}$$

where k is the spatial wavenumber and k_c is the characteristic cut-off wavenumber.

Cauchy filter in physical space

The Cauchy filter has the form:

$$G(r) = G(x_1, x_2, x_3) = \frac{a}{\pi\Delta[(x_1^2 + x_2^2 + x_3^2)/\Delta^2 + a^2]}, a = \pi/24 \, . \tag{2.63}$$

2.6.2 Mass-weighted Favre-filtering

A mass-weighted Favre-filter for any quantity f is defined as

$$\overline{\rho}\widetilde{f}(x,t) = \int_{-\infty}^{\infty} \rho f(r,t) G(r,x) dr \, , \tag{2.64}$$

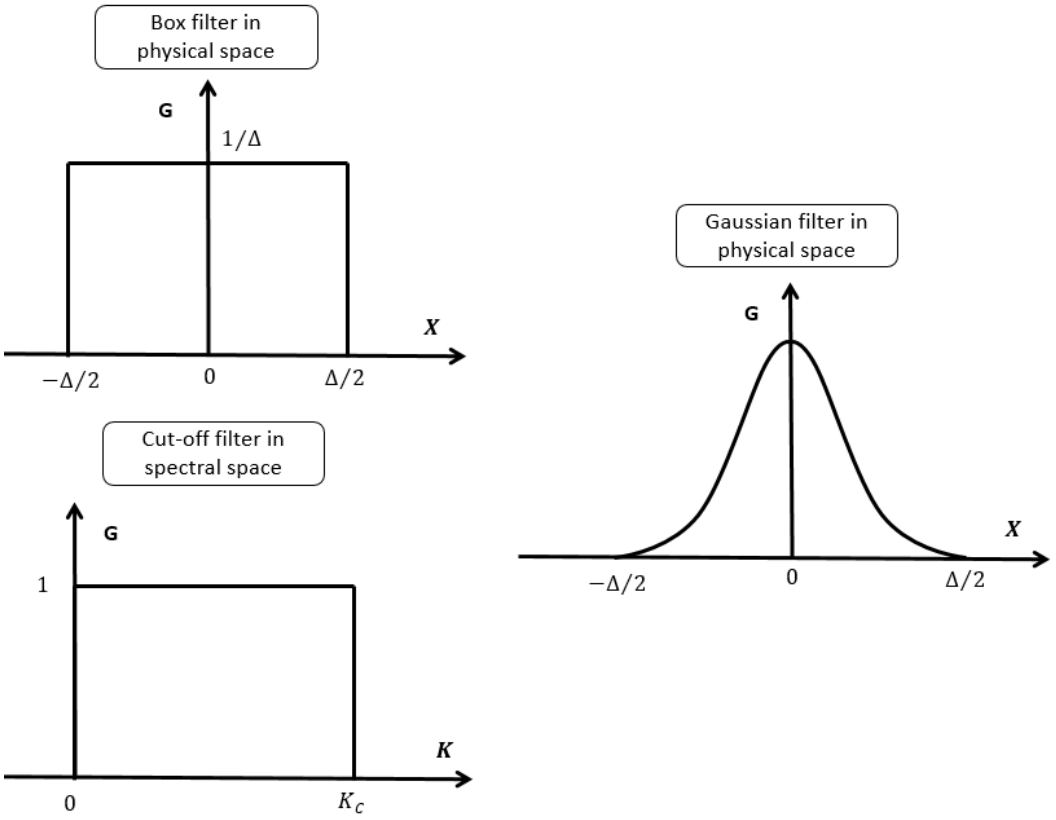

FIGURE 2.3: Common LES filter functions, as adapted from Poinsot et al. [71].

where ρ denotes the density of the fluid and $\widetilde{f}(x,t)$ represents the resolved part of f in the numerical simulations, which is calculated as follows (Favre filter):

$$\widetilde{f} = \frac{\overline{\rho f}}{\overline{\rho}} \tag{2.65}$$

2.6.3 Favre-filtered Navier-Stokes equations

The Favre-filtered form of Navier-Stokes equations are shown below. Further information and mathematical derivation can be found in the book from Poinsot [71].
Conservation of mass:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_i}\left(\overline{\rho}\widetilde{u}_i\right) = 0 \,. \tag{2.66}$$

Conservation of momentum:

$$\frac{\partial \overline{\rho}\widetilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{u}_i\widetilde{u}_j\right) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\overline{\tau}_{ij} - \overline{\rho}(\widetilde{u_i u_j} - \widetilde{u}_i\widetilde{u}_j)\right] \,. \tag{2.67}$$

Conservation of species:

$$\frac{\partial}{\partial t}(\overline{\rho}\widetilde{Y}_k) + \frac{\partial}{\partial x_i}(\overline{\rho}\widetilde{Y}_k\widetilde{u}_i) = \frac{\partial}{\partial x_i}\left[\overline{V_{k,i}Y_k} - \overline{\rho}\left(\widetilde{u_iY_k} - \widetilde{u}_i\widetilde{Y_k}\right)\right] + \overline{\dot{\omega}}_k \ . \tag{2.68}$$

Conservation of energy:

$$\begin{aligned}\frac{\partial \overline{\rho}\widetilde{e}}{\partial t} + \frac{\partial}{\partial x_i}(\overline{\rho}\widetilde{e}\widetilde{u}_i) = \frac{\partial}{\partial x_i}\left[\overline{\lambda\frac{\partial T}{\partial x_i}} - \overline{\rho}(\widetilde{u_ie} - \widetilde{u}_i\widetilde{e})\right] + \frac{\partial}{\partial x_i}(\widetilde{\sigma_{ij}u_j}) \\ + \frac{\partial}{\partial x_i}\left[\widetilde{\sigma_{ij}u_j} - \widetilde{\sigma_{ij}}\widetilde{u}_j\right] - \frac{\partial}{\partial x_i}\left(\overline{\rho\sum_{k=1}^{N} V_{i,k}Y_kh_{s,k}}\right) + \overline{\dot{\omega}}_T \ .\end{aligned} \tag{2.69}$$

In Eqs. 2.66-2.69, there are several unresolved terms for which mathematical modelling is required:

- Unresolved stresses: $\left(\widetilde{u_iu_j} - \widetilde{u}_i\widetilde{u}_j\right)$
- Unresolved species fluxes: $\left(\widetilde{u_iY_k} - \widetilde{u}_i\widetilde{Y_k}\right)$
- Unresolved energy fluxes: $\left(\widetilde{u_ie} - \widetilde{u}_i\widetilde{e}\right)$
- Unresolved viscous diffusion: $\frac{\partial}{\partial x_i}\left(\widetilde{\sigma_{ij}u_j} - \widetilde{\sigma_{ij}}\widetilde{u}_j\right)$ (this term is quite small and can be neglected [83])
- Filtered laminar diffusion fluxes $\overline{V_{k,i}Y_k} = -\overline{\rho}\overline{D}_k(\partial\widetilde{Y}_k/\partial x_i)$
- Filtered temperature gradient: $\overline{\lambda(\partial T/\partial x_i)} = \overline{\lambda}(\partial\widetilde{T}/\partial x_i)$
- Filtered chemical reaction rate $\overline{\dot{\omega}}_k$
- Filtered heat release rate $\overline{\dot{\omega}}_T$

2.6.4 Modelling of unresolved stresses

The unresolved stresses represent the dynamic coupling between the large and the small scales in the turbulence. In this part, several approaches to model the unresolved Reynolds stresses in non-reacting flows are discussed, for instance, eddy viscosity models [84, 85, 86, 87] possibly relying on the Germano identity and dynamic model [88, 89, 90, 91], or similarity, nonlinear and deconvolution models [92, 93, 94, 95].

Smagorinsky model

In the LES context, the Smagorinsky model [84] is widely used to compute the unresolved

stresses assuming isotropic turbulence. Using the Boussinesq assumption, the unresolved stresses $\tau_{ij} = (\widetilde{u_i u_j} - \widetilde{u}_i \widetilde{u}_j)$ of the incompressible fluid can be computed as follows:

$$\tau_{ij} - \frac{\delta_{ij}}{3}\tau_{kk} = -\nu_t\left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial \tilde{u}_k}{\partial x_k}\right) = -2\nu_t\left(\widetilde{S}_{ij} - \frac{\delta_{ij}}{3}\widetilde{S}_{kk}\right) , \tag{2.70}$$

where the turbulent viscosity ν_t is computed from [71]:

$$\nu_t = C_S^2 \Delta^{4/3} l_0^{2/3} (2\tilde{S}_{ij}\tilde{S}_{ij})^{1/2} . \tag{2.71}$$

Here, l_0, C_S and $\tilde{S}_{ij}$ are the turbulence integral length scale, the model constant and the resolved shear stress, respectively. Assuming $l_0 \approx \Delta$, the turbulent viscosity becomes:

$$\nu_t = (C_S\Delta)^2 \left|\tilde{S}\right| = (C_S\Delta)^2 (2\tilde{S}_{ij}\tilde{S}_{ij})^{1/2} . \tag{2.72}$$

Lily et al. [96, 97] suggested the value of the model constant C_S to be in the range of 0.17-0.21 based on the analysis of the decay rates of isotropic turbulent eddies in the inertial subrange of the energy spectrum. A review by Rogallo et al. [98] suggested values of 0.19-0.24 for C_S.

For the compressible fluid, the sub-grid energy τ_{ij} and τ_{kk} are modeled separately [84]:

$$\tau_{ij} - \frac{\delta_{ij}}{3}\tau_{kk} = -C_S^2 2\Delta^2 \overline{\rho} \left|\tilde{S}\right| \left(\widetilde{S}_{ij} - \frac{\delta_{ij}}{3}\widetilde{S}_{kk}\right) \ , \tau_{kk} = C_I 2\Delta^2 \overline{\rho} \left|\tilde{S}\right|^2 \tag{2.73}$$

In Eq. 2.73, the model constant C_I is chosen as $C_I = 0.009$ for the calculation of the trace τ_{kk} of the stress tensor.

Sigma model

In the Sigma model proposed by Nicoud et al. [87], the sub-grid turbulent viscosity ν_t is the function of a differential operator D_σ, the integral length-scale $l_0 \approx \Delta$ and a model constant C_σ.

The differential operator is defined as:

$$D_\sigma = \frac{\sigma_3(\sigma_1 - \sigma_2)(\sigma_2 - \sigma_3)}{\sigma_1^2} . \tag{2.74}$$

The turbulent viscosity is outlined as below:

$$\nu_t = (C_\sigma \Delta)^2 D_\sigma \ . \tag{2.75}$$

In Eqs. 2.74 and 2.75, σ_1, σ_2, and σ_3 are three singular values of the resolved velocity gradient tensor $\widetilde{g}_{ij} = \partial \widetilde{u}_i / \partial x_j$, where $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0$ and can be calculated from the invariants of the matrix $\mathcal{G}_{ij} = \widetilde{g}_{ki}\widetilde{g}_{kj}$.

The invariants of $\boldsymbol{\mathcal{G}}$ are computed from:

$$\begin{aligned} I_1 &= \mathrm{tr}(\boldsymbol{\mathcal{G}}) \ , \\ I_2 &= \frac{1}{2}\big(\mathrm{tr}(\boldsymbol{\mathcal{G}})^2 - \mathrm{tr}(\boldsymbol{\mathcal{G}}^2)\big) \ , \\ I_3 &= \det(\boldsymbol{\mathcal{G}}) \ . \end{aligned} \tag{2.76}$$

The angles α_i are computed from the invariants I_1, I_2, I_3:

$$\begin{aligned} \alpha_1 &= \frac{I_1^2}{9} - \frac{I_2}{3} \ , \\ \alpha_2 &= \frac{I_1^3}{27} - \frac{I_1 I_2}{6} + \frac{I_3}{2} \ , \\ \alpha_3 &= \frac{1}{3}\arccos\frac{\alpha_2}{\alpha_1^{3/2}} \ . \end{aligned} \tag{2.77}$$

The singular values are calculated as:

$$\begin{aligned} \sigma_1 &= \Big(\frac{I_1}{3} - 2\sqrt{\alpha_1}\cos\alpha_3\Big)^{1/2} \ , \\ \sigma_2 &= \Big(\frac{I_1}{3} - 2\sqrt{\alpha_1}\cos\big(\frac{\pi}{3} + \alpha_3\big)\Big)^{1/2} \ , \\ \sigma_3 &= \Big(\frac{I_1}{3} - 2\sqrt{\alpha_1}\cos\big(\frac{\pi}{3} - \alpha_3\big)\Big)^{1/2} \ . \end{aligned} \tag{2.78}$$

Germano dynamic model

Considered as an improvement of the Smagorinsky approach, the Germano dynamic model [88, 89] introduced a different decomposition of the unresolved turbulent stresses using two filter operations with two filter sizes of $\overline{\Delta}$ and $\widehat{\Delta}$ ($\overline{\Delta} < \widehat{\Delta}$).

The unresolved stresses with LES filter size $\overline{\Delta}$ read:

$$\tau_{ij}^I = \big(\overline{u_i u_j} - \overline{u}_i \overline{u}_j\big) \ . \tag{2.79}$$

The unresolved stresses at the second filter level read:

$$\tau_{ij}^{II} = \left(\widehat{\overline{u_i u_j}} - \widehat{\overline{u}}_i \widehat{\overline{u}}_j\right) . \tag{2.80}$$

The Germano identity $\mathcal{L}_{ij}$ is introduced using τ_{ij}^{I} and τ_{ij}^{II}:

$$\mathcal{L}_{ij} = \left(\widehat{\overline{u}_i \overline{u}_j} - \widehat{\overline{u}}_i \widehat{\overline{u}}_j\right) = \tau_{ij}^{II} - \widehat{\tau}_{ij}^{I} . \tag{2.81}$$

Using the Smagorinsky model, τ_{ij}^{I} and τ_{ij}^{II} are computed from:

$$\tau_{ij}^{I} - \frac{\delta_{ij}}{3}\tau_{kk}^{I} = -2C_{sgs}^2 \overline{\Delta}^2 |\overline{S}| \overline{S}_{ij} , \tag{2.82}$$

$$\tau_{ij}^{II} - \frac{\delta_{ij}}{3}\tau_{kk}^{II} = -2C_{sgs}^2 \widehat{\Delta}^2 |\widehat{\overline{S}}| \widehat{\overline{S}}_{ij} . \tag{2.83}$$

Equation 2.81 can be rewritten as

$$\mathcal{L}_{ij} - \frac{\delta_{ij}}{3}\mathcal{L}_{kk} = C_{sgs}^2 M_{ij} , \tag{2.84}$$

where M_{ij} has the form:

$$M_{ij} = 2\overline{\Delta}^2 \widehat{|\overline{S}| \overline{S}_{ij}} - 2\widehat{\Delta}^2 |\widehat{\overline{S}}| \widehat{\overline{S}}_{ij} . \tag{2.85}$$

Following a suggestion by Lily et al. [90], the local value of C_{sgs} can be computed as:

$$C_{sgs}^2 = \frac{\langle \mathcal{L}_{ij} M_{ij} \rangle}{\langle M_{ij} M_{ij} \rangle} . \tag{2.86}$$

In Eq. 2.86, the brackets $\langle\rangle$ imply an averaging operator. Several approaches for the determination of C_{sgs} can also be found in the work of Bardina et al. [92], Lesieur et al. [85] or Meneveau et al. [99].

2.6.5 Modelling of unresolved scalar transport

Using a gradient assumption, the calculation of the unresolved scalar fluxes is:

$$\widetilde{u_i Y_k} - \widetilde{u}_i \widetilde{Y_k} = -\frac{\nu_t}{Sc_t} \frac{\partial \widetilde{Y}_k}{\partial x_i} , \tag{2.87}$$

where Sc_t denotes a turbulent Schmidt number. It is noted that the counter-gradient fluxes must also be considered in the flame surface density approach for combustion modelling (see section 2.7).

2.6.6 Unresolved energy fluxes

Considering the definition of the total energy $e = 1/2(u_i u_i + c_v T)$ from Eq. 2.6, the unresolved energy fluxes can be rewritten as:

$$\begin{aligned} \widetilde{u_i e} - \widetilde{u}_i \widetilde{e} &= \left(\frac{1}{2}\widetilde{u_i u_j u_j} + c_v \widetilde{u_i T}\right) - \left(\frac{1}{2}\widetilde{u}_i \widetilde{u_j u_j} + c_v \widetilde{u}_i \widetilde{T}\right) \\ &= \frac{1}{2}\left(\widetilde{u_i u_j u_j} - \widetilde{u}_i \widetilde{u_j u_j}\right) + c_v\left(\widetilde{u_i T} - \widetilde{u}_i \widetilde{T}\right) . \end{aligned} \tag{2.88}$$

The rewritten form of the unresolved energy fluxes introduces two unresolved terms $1/2\left(\widetilde{u_i u_j u_j} - \widetilde{u}_i \widetilde{u_j u_j}\right)$ and $c_v\left(\widetilde{u_i T} - \widetilde{u}_i \widetilde{T}\right)$.

Using a model proposed by Knight et al. [100], the subfilter-scale turbulent transport can be estimated as follows:

$$\frac{1}{2}\left(\widetilde{u_i u_j u_j} - \widetilde{u}_i \widetilde{u_j u_j}\right) = \tau_{ij} \widetilde{u}_i . \tag{2.89}$$

The second unresolved term can also be computed using the standard gradient assumption:

$$c_v\left(\widetilde{u_i T} - \widetilde{u}_i \widetilde{T}\right) = -c_v \frac{\nu_t}{\mathrm{Pr}_t} \frac{\partial \widetilde{T}}{\partial x_i} . \tag{2.90}$$

In Eq. 2.90, Pr_t denotes the turbulent Prandtl number.

2.7 Combustion modelling using flame surface density

Modelling of reaction rates in turbulent premixed combustion is challenging since the thickness of the flame is typically within 0.1 mm to 1 mm, which is usually much smaller than the LES filter size Δ. Additionally, at the flame profile, the thermochemical variables change very sharply, which can lead to numerical instabilities. To circumvent the difficulty, three approaches have been proposed: (1) simulation of an artificial thickened flame; (2) solving the G-equation to track the flame front; or (3) use of a filtered progress

variable with a reaction rate that can be directly estimated from the flame surface density (FSD) [71]. In the scope of this work, the interaction between the turbulence and the chemistry is modeled by the FSD approach, where the flamelet assumption is used to obtain the relation between the turbulent burning rate and the flame area or surface density. The flame surface density can be determined either through an algebraic model or a modelled transport equation.

2.7.1 Transport equation for the progress variable

In the FSD approach, the flame propagation is described by the progress variable c [101], which is based on the fuel mass fraction Y_F, where Y_F^u denotes the unburned and Y_F^b the burned state ($Y_F^u = 0$ for lean condition).

$$c = \frac{Y_F - Y_F^u}{Y_F^b - Y_F^u} \tag{2.91}$$

The transport equation for the Favre-filtered progress variable reads:

$$\frac{\partial \overline{\rho}\widetilde{c}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{c}\widetilde{u}_i}{\partial x_i} + \frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu}_i - \widetilde{c}\,\widetilde{u}_i)] = \overline{\frac{\partial}{\partial x_i}\left(\rho D \frac{\partial c}{\partial x_i}\right)} + \overline{\dot{w}}_c = \langle \rho S_d \rangle_S \, \Sigma_{\text{gen}} \, . \tag{2.92}$$

In this equation, $\overline{\dot{w}}_c$, D, S_d and Σ_{gen} are the mean reaction source term, the molecular diffusivity, the displacement speed and the generalized flame surface density, respectively. The generalized FSD approach in principle combines molecular diffusion and the source term.

The modelling of the sub-grid scalar flux is shown in the following equation [102]:

$$\frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu}_i - \widetilde{c}\widetilde{u}_i)] = -\frac{\partial}{\partial x_i}\left(\overline{\rho}\frac{\nu_t}{S_c}\frac{\partial \widetilde{c}}{\partial x_i}\right) + \frac{\partial}{\partial x_i}[\rho_u s_L(\overline{c} - \widetilde{c})] \tag{2.93}$$

In Eq. 2.93, the turbulent viscosity, the Schmidt number, the unburned density and the un-stretched laminar flame speed are represented by μ_t, S_c, ρ_u and s_L. The second term on the RHS of Eq. 2.93 is referred to as the counter gradient (F_{cgt}). The terms $\overline{c}$ and $\widetilde{c}$ are related by the heat release rate τ where $\overline{c} = (1+\tau)\widetilde{c}/(1+\tau\widetilde{c})$ [31].

The generalized flame surface density Σ_{gen} can be approximately computed as follows:

$$\begin{aligned} \langle \rho S_d \rangle_S \Sigma_{\text{gen}} \approx \rho_u s_L \Sigma_{\text{gen}} &= \rho_u s_L \Xi |\nabla \bar{c}| - F_{cgt} \\ &\approx \rho_u s_L \Xi |\nabla \widetilde{c}| \ . \end{aligned} \tag{2.94}$$

In Eq. 2.94, Ξ and F_{cgt} represents the wrinkling factor and the counter gradient, respectively. As it has been shown by Ma et al. [103], the counter gradient transport F_{cgt} is implicitly modelled by replacing the Reynolds-averaged by the Favre-filtered progress variable in Eq. 2.94. The generalized flame surface density Σ_{gen} in Eq. 2.92 and 2.94 can be determined using two different approaches: algebraic models and transport equation.

2.7.2 Algebraic models

Algebraic models are convenient approaches to model the FSD instead of solving a transport equation with many unclosed terms. In the algebraic models, the influence of turbulence on the dynamics of the flame is incorporated via the sub-grid turbulent kinetic energy (or the sub-grid turbulent fluctuation). In the scope of thesis, algebraic models of FSD are used.

Some existing algebraic models for FSD

Algebraic models for FSD have been developed and have been widely used for the determination of the flame surface density Σ_{gen}. Several popular models are introduced as follows.

In a model proposed by Charlette et al. [38], the flame surface density is calculated based on the filtered (unweighted) progress variable $\bar{c}$:

$$\Sigma_{\text{gen}} = \left[1 + \min\left[\frac{\Delta}{\delta_L}, \Gamma_\Delta\left(\frac{u'}{s_L}\right)\right]\right]^\beta |\nabla \bar{c}| \ , \beta = 0.5 \tag{2.95}$$

In Eq. 2.95, Γ_Δ represents for the intermittent turbulence net flame stretch (ITNFS) model as shown in the following equations:

$$\begin{aligned}
\Gamma_\Delta &= \left[\left(\left(f_u^{-a} + f_\Delta^{-a}\right)^{-1/a}\right)^{-b} + f_{\mathrm{Re}}^{-b}\right] , \\
f_u &= 4\left(\frac{27}{110}C_k\right)^{1/2}\left(\frac{18}{55}C_k\right)\left(\frac{u'}{s_L}\right)^2 \\
C_k &= 1.5 \, , f_\Delta = \left[\frac{27C_k\pi^{4/3}}{110}\left(\left(\frac{\Delta}{\delta_L}\right)^{4/3} - 1\right)\right]^{1/2} \\
f_{\mathrm{Re}} &= \left[\frac{9}{55}\exp\left(-\frac{3}{2}C_k\pi^{4/3}\mathrm{Re}_\Delta^{-1}\right)\right]^{1/2}\mathrm{Re}_\Delta^{1/2} \, , \mathrm{Re}_\Delta = 4\frac{\Delta}{\delta_L}\frac{u'}{s_L} \\
a &= 0.6 + 0.2\exp[-0.1(u'/s_L)] - 0.2\exp[-0.01(\Delta/\delta_L)] \, , b = 1.4
\end{aligned} \tag{2.96}$$

A quite similar approach for the modelling of the flame surface density (as in Eq. 2.95) can be found in the model by Angelberger et al. [104]:

$$\Sigma_{\mathrm{gen}} = \left[1 + \alpha\Gamma_{\mathrm{k}}\left(\frac{u'}{s_L}\right)\right]|\nabla\bar{c}| \tag{2.97}$$

In equation 2.97, α is a constant and the efficiency function Γ_{n} is calculated as follows:

$$\Gamma_{\mathrm{n}} = 0.75\exp\left[-\frac{1.2}{(u'/s_L)^{0.3}}\right]\left(\frac{\Delta}{\delta_L}\right)^{2/3} \tag{2.98}$$

A simple model proposed by Borger et al. [105] reads:

$$\Sigma_{\mathrm{gen}} = K_\Sigma\frac{\bar{c}(1-\bar{c})}{\Delta} \, . \tag{2.99}$$

The value of the model constant K_Σ in Eq. 2.99 is estimated to be $4\sqrt{6/\pi}$.

Based on the experimental data of a methane Bunsen flame, Muppala et al. [32] developed a model for the wrinkling factor Ξ, where the effect of the pressure variation on the flame propagation is considered as shown below.

$$\Xi = 1 + a\mathrm{Re}_t^{0.25}\left(\frac{u'}{s_L}\right)^b\left(\frac{p}{p_0}\right)^c \tag{2.100}$$

The turbulent Reynolds number Re_t is computed as $\mathrm{Re}_t = u'\Delta/\nu$. For the iso-octane/air flame, the model parameters are set to $b = 0.2$, $c = 0.2$, and $a = 0.46/\mathrm{Le}$ [32, 106] (Le is the Lewis number).

The sub-grid scale velocity fluctuation u' is estimated according to Wyngaard [107]:

$$u' = \sqrt{\frac{2}{3} k_{sgs}} \ , \ \ k_{sgs} = \frac{1}{(C_v \Delta)} \nu_t^2 \ . \tag{2.101}$$

Here the constant C_v is chosen to be 0.1 [108]. More elaborate discussions on the fundamental theory of FSD can be found in the work of Marble et al. [28], Pope [29], Bray et al. [31], Candel et al. [109], Vervisch et al. [110], or the PhD theses from Hawkes [111] and Lin [112], amongst others.

FSD in the context of engine simulations

In the context of engine simulations, the gas density, temperature and the in-cylinder pressure vary during the engine operation. Therefore, in Eq. 2.94, the unburned gas density ρ_u is evaluated as a function of the ambient pressure p_0 and density ρ_0, the heat capacity ratio γ and the actual pressure p inside the combustion chamber.

$$\rho_u = \rho_0 \big(p/p_0\big)^{(1/\gamma)} \ . \tag{2.102}$$

Considering the effect of the unburned temperature and the actual in-cylinder pressure, the laminar flame speed (see Eq. 2.94) can be determined as follows [106]:

$$s_L = s_{L,0} \big(\frac{T_u}{T_0}\big)^{\alpha} \big(\frac{p}{p_0}\big)^{\beta} \ . \tag{2.103}$$

In Eq. 2.103, the unburned gas temperature T_u can be computed from the isentropic process $T_u = T_0 \big(p/p_0\big)^{\frac{\gamma-1}{\gamma}}$. The laminar flame speed at ambient condition $s_{L,0}$ and the two parameters α and β depend on the specific fuel, the equivalence ratio and the burned gas dilution fraction [106].

To start the combustion, the ignition kernel is created in a small area (flame kernel) near the spark-plug by imposing the value 1.0 for the progress variable c. Depending on the flammability limits of the fuel, the minimum ignition energy and critical radius of the initial flame kernel can be estimated according to the book of Turns [113].

Disadvantage of algebraic models for FSD

Although algebraic models of FSD are simple to implement and easy to apply for different fuels, the main issue of algebraic approaches is that they are only suitable in situations, where the production and dissipation of FSD are locally in equilibrium at the flame

front. In turbulent combustion, the process of flame propagation is strongly influenced by complicated interactions between many effects such as the mean flow- and sub-grid convection, the fluid strain, the planar propagation, the curved-flame propagation, the sub-grid destruction of flame surface area and the production or destruction of FSD. These effects vary spatially and temporally, which cannot be properly described by simple algebraic approaches. In particular, these assumptions do not hold during early flame propagation, right after ignition. Hence, solving the transport equation for FSD is necessary for proper modelling of turbulent combustion. A detailed analysis on this issue can be found in the work of Hawkes et al. [114].

2.7.3 Transport equation for FSD

Instead of using some simple algebraic models, the flame surface density Σ_{gen} can be obtained by solving a transport equation. The Favre-filtered transport equation for the FSD is formulated as [112]:

$$\frac{\partial}{\partial t}\left(\overline{\rho}\widetilde{\Sigma}\right)+\frac{\partial}{\partial x_i}\left(\overline{\rho}\widetilde{\Sigma}\widetilde{u}_i\right)+\frac{\partial}{\partial x_i}\overline{\rho}\widetilde{\Sigma}\widetilde{u}_i\left[\overline{(u_i)_{\text{s}}}-\widetilde{u}_i\right]=\overline{(\alpha_{\text{T}})_{\text{s}}}\overline{\rho}\widetilde{\Sigma}-\frac{\partial}{\partial x_i}\left[\overline{(u_{\text{p}}N_i)_{\text{s}}}\overline{\rho}\widetilde{\Sigma}\right]+\overline{\left(u_{\text{p}}\frac{\partial N_i}{\partial x_i}\right)_{\text{s}}}\overline{\rho}\widetilde{\Sigma}\quad . \tag{2.104}$$

In this transport equation for the FSD, the term $\left[\overline{(u_i)_{\text{s}}}-\widetilde{u}_i\right]$ represents the sub-grid flux [30] and $\mathbf{N}=-\nabla c/|\nabla c|$ is the normal vector of the local flame pointing towards the reactants. The term $\overline{(\alpha_{\text{T}})_{\text{s}}}$ corresponds to the strain effect of the surrounding fluid on the flame surface:

$$\overline{(\alpha_{\text{T}})_{\text{s}}}=\overline{\left[(\delta_{ij}-N_iN_j)\frac{\partial u_i}{\partial x_j}\right]_{\text{s}}}\quad . \tag{2.105}$$

For further details on the modeling of $\overline{(\alpha_{\text{T}})_{\text{s}}}$, readers are referred to the studies of Cant et al. [30], Yeung et al. [115], Trouve et al. [116] or Ma et al. [103].

The terms $\partial\left[\overline{(u_{\text{p}}N_i)_{\text{s}}}\overline{\rho}\widetilde{\Sigma}\right]/\partial x_i$ and $\overline{\left(u_{\text{p}}(\partial N_i/\partial x_i)\right)}\overline{\rho}\widetilde{\Sigma}$ describe the propagation, production and destruction of FSD under the influence of the flame propagation and curvature. The modeling approaches of these terms are provided in a study of of Pope [29] or in the PhD theses from Hawkes [111], Lin [112] or Ma et al. [103].

Chapter 3

Numerical treatment and implementation

In this chapter, the finite volume method for a convection-diffusion problem is introduced along with several commonly used numerical schemes for the spatial and temporal discretization. A detailed derivation of non-reflecting boundaries for the compressible fluid is presented. To provide an insight on grid generation, an approach using voxels and voxelization to generate an equidistant Cartesian grid for engine geometries is discussed. As it is one of the main issue in engine simulation, an efficient technique using Lagrangian particles and immersed boundaries to numerically describe the moving objects are thoroughly explained.

3.1 Finite volume method for convection-diffusion problems

The finite volume method (FVM) is a technique that is commonly employed for the discretization of partial differential equations, in which the volume integral formulation is used on a finite partitioning set of control volumes. In this section, the discretization of a transport equation using FVM for the convection $\int\limits_{\Delta V} \frac{\partial \Phi u_j}{\partial x_j} \mathrm{d}V$, diffusion $\int\limits_{\Delta V} \frac{\partial D_j(\Phi)}{\partial x_j} \mathrm{d}V$ and source terms $\int\limits_{\Delta V} S_\Phi \mathrm{d}V$ is presented. This section closely follows the PhD thesis of Kempf [117].

The integral form of a transport equation for a scalar Φ over a volume ΔV reads:

$$\int_{\Delta V} \frac{\partial \Phi}{\partial t} \mathrm{d}V + \int_{\Delta V} \frac{\partial \Phi u_j}{\partial x_j} \mathrm{d}V = \int_{\Delta V} \frac{\partial D_j(\Phi)}{\partial x_j} \mathrm{d}V + \int_{\Delta V} S_\Phi \mathrm{d}V \ . \tag{3.1}$$

For the convection, Gauss's theorem to transform the convective and diffusive terms in Eq. 3.1 into fluxes over surfaces $\partial \Delta V$ is used:

$$\int_{\Delta V} \frac{\partial \Phi u_j}{\partial x_j} \mathrm{d}V = \int_{\Delta A} \Phi n_j \mathrm{d}A \ , \tag{3.2}$$

and for the diffusive terms:

$$\int_{\Delta V} \frac{\partial D_j(\Phi)}{\partial x_j} \mathrm{d}V = \int_{\Delta A} D_j(\Phi) n_j \mathrm{d}A \ . \tag{3.3}$$

Here, the equation 3.1 can be rewritten as

$$\frac{\partial}{\partial t} \int_{\Delta V} \big(\Phi \mathrm{d}V\big) + \int_{\Delta A} \big(\Phi u_j\big) n_j \mathrm{d}A = \int_{\Delta A} D_j(\Phi) n_j \mathrm{d}A + \int_{\Delta V} S_\Phi \mathrm{d}V \ . \tag{3.4}$$

Here, by introducing the means $\Phi^{\Delta V}$ of Φ and $S^{\Delta V}$ of the source term S_Φ, Eq. 3.4 can be adapted to Eq. 3.5 for one cell of the volume ΔV:

$$\frac{\partial}{\partial t}\big(\Phi^{\Delta V} \delta V\big) + \sum_{u,d,l,r,t,b} F_{f,\mathrm{C}} = \sum_{u,d,l,r,t,b} F_{f,\mathrm{D}} + S^{\Delta V}\big(\Phi\big)\delta V \ . \tag{3.5}$$

In Eq. 3.5, the convective flux $F_{f,\mathrm{C}}$ and the diffusion flux $F_{f,\mathrm{D}}$ are fluxes across the surfaces of the neighboring control volumes, i.e., the upstream $\mathrm{CV_U}$, downstream $\mathrm{CV_D}$, left $\mathrm{CV_L}$, right $\mathrm{CV_R}$, top $\mathrm{CV_T}$ and the bottom $\mathrm{CV_B}$, as illustrated in Fig. 3.1.

To fully evaluate all the terms in Eq. 3.5, the discretization of volume and surface integrals (Eqs. 3.6 and 3.7) are necessary.

$$\int_{\Delta V} \Phi \mathrm{d}V = \Phi^{\Delta V} \Delta V \approx \Phi_\mathrm{C} \Delta V \ , \tag{3.6}$$

$$\int_{\Delta A_f} \Phi \mathrm{d}A_f = \Phi^{\Delta A_f} \Delta A_f \approx \Phi_f \Delta A_f \ . \tag{3.7}$$

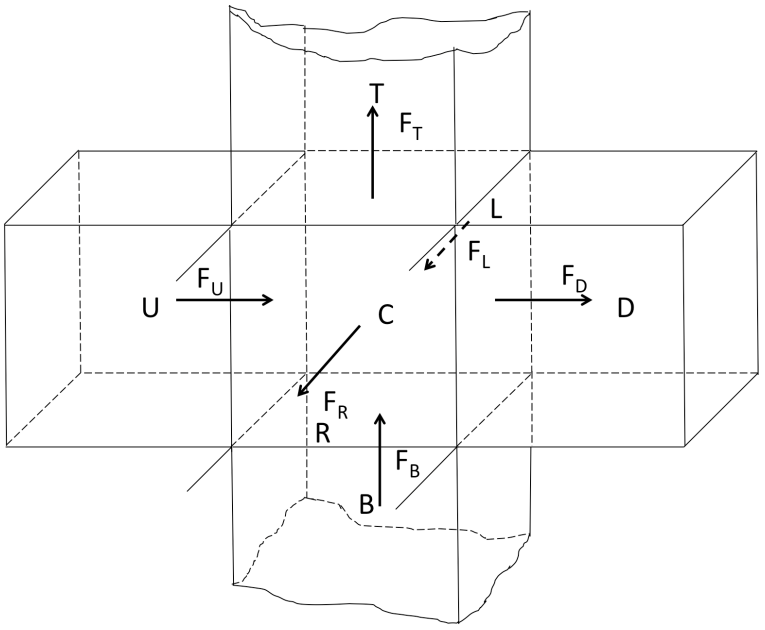

FIGURE 3.1: Illustration of the fluxes over a control volume ΔV.

Using Eqs. 3.6 and 3.7, the convective and the diffusive fluxes are determined from

$$\int_{\Delta A} (\Phi u_j) n_j \mathrm{d}A = \sum_{u,d,l,r,t,b} F_{f,\mathrm{C}} \quad , \tag{3.8}$$

and the flux over the surface ΔA_f is given as:

$$F_{f,\mathrm{C}} \approx (\Phi u_j)_f n_j \Delta A_f \tag{3.9}$$

In the orthogonal grids, the convective flux can be approximated from the value at the center of the cell face Φ_f and the surface normal component u_f:

$$F_{f,\mathrm{C}} \approx \Phi_f u_f \Delta A_f \tag{3.10}$$

The discretization of the diffusive flux is computed in a similar way:

$$\int_{\Delta A} D_j(\Phi) n_j \mathrm{d}A = \sum_{u,d,l,r,t,b} F_{f,\mathrm{D}} \tag{3.11}$$

3.2 Spatial discretization schemes

The discretization of a partial differential equation in time or space requires numerical schemes. In the scope of this thesis, some commonly used numerical schemes are introduced including Upwind Differencing Scheme (UDS), Central Differencing Scheme (CDS), Quadratic Upwind Interpolation for Convective Kinematics (QUICK), Total Variation Diminishing (TVD), Crank-Nicolson and Runge-Kutta schemes.

3.2.1 Central differencing scheme (CDS)

In the central differencing scheme of second order, the values of the physical quantities at the cell faces are calculated by a linear interpolation. The cell face values are computed with a 2^{nd} order CDS in a uniform grid (Δ=const) (see Fig. 3.2) as follows:

$$\begin{aligned} \Phi_{\text{w}} &= \frac{\Phi_{\text{U}} + \Phi_{\text{C}}}{2} \quad , \\ \Phi_{\text{e}} &= \frac{\Phi_{\text{C}} + \Phi_{\text{D}}}{2} \quad . \end{aligned} \tag{3.12}$$

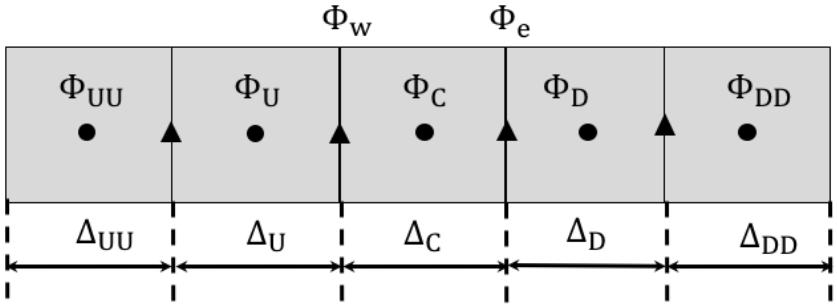

FIGURE 3.2: Illustration of an one dimensional computational grid.

In case of using a 4^{th} order CDS, the calculations of Φ_{w} and Φ_{e} will require more cells with chosen weighting factors, for instance:

$$\begin{aligned} \Phi_{\text{w}} &= \frac{-3}{48}\Phi_{\text{UU}} + \frac{27}{48}\Phi_{\text{U}} + \frac{27}{48}\Phi_{\text{C}} + \frac{-3}{48}\Phi_{\text{D}} \quad , \\ \Phi_{\text{e}} &= \frac{-3}{48}\Phi_{\text{U}} + \frac{27}{48}\Phi_{\text{C}} + \frac{27}{48}\Phi_{\text{D}} + \frac{-3}{48}\Phi_{\text{DD}} \quad . \end{aligned} \tag{3.13}$$

Applying the CDS to the discretization of the convection or the diffusion fluxes may cause oscillations in the numerical results and could also lead to convergence problems.

3.2.2 Upwind differencing scheme

In the upwind differencing scheme, the transported quantities at the cell faces Φ_e and Φ_w are copied from the cell-center value of the up-stream or down-stream cell, depending on the flow direction. The quantities at the cell faces are determined by the first order UDS as:

$$\begin{aligned} \Phi_{\mathrm{w}} &= \Phi_{\mathrm{U}} \\ \Phi_{\mathrm{e}} &= \Phi_{\mathrm{C}} \end{aligned} \tag{3.14}$$

Using a 2^{nd} order UDS, the formulation leads to:

$$\begin{aligned} \Phi_{\mathrm{w}} &= \frac{3}{2}\Phi_{\mathrm{U}} - \frac{1}{2}\Phi_{\mathrm{UU}} \\ \Phi_{\mathrm{e}} &= \frac{3}{2}\Phi_{\mathrm{C}} - \frac{1}{2}\Phi_{\mathrm{U}} \end{aligned} \tag{3.15}$$

Achieving only the first order of accuracy, UDS is very dissipative but unconditionally stable. In simulations of turbulent flows, applying USD for the discretization of the convection will destroy the small structures of the flow field, therefore, it should be avoided.

3.2.3 Quadratic upwind interpolation for convective kinematics (QUICK)

In comparison to a 2^{nd} order UDS scheme, a QUICK scheme [118] employs an upstream-weighted quadratic interpolation of two upstream nodes and one down stream node for the calculation of the quantities at the cell faces.

$$\begin{aligned} \Phi_{\mathrm{w}} &= \frac{6}{8}\Phi_{\mathrm{U}} + \frac{3}{8}\Phi_{\mathrm{C}} - \frac{1}{8}\Phi_{\mathrm{UU}} \\ \Phi_{\mathrm{e}} &= \frac{6}{8}\Phi_{\mathrm{C}} + \frac{3}{8}\Phi_{\mathrm{D}} - \frac{1}{8}\Phi_{\mathrm{U}} \end{aligned} \tag{3.16}$$

With a third order of accuracy on a uniform grid [119], a QUICK scheme could be a better choice than a 2^{nd} order CDS or UDS. However, using QUICK can be problematic, because the main interpolation coefficients, e.g., 6/8 or 3/8 are not always positive since they are depending on the flow direction, while the coefficients of Φ_{UU} or Φ_{DD}, e.g., $-1/8$ are negative. This may lead to stability problems and unbounded solutions under

complicated conditions [119]. As a matter of fact, the QUICK scheme is conditionally stable.

3.2.4 Total variation diminishing (TVD)

There are common issues with the previously presented numerical schemes. Low order schemes are dissipative whilst high order schemes often lead to oscillation and convergence problems. Obviously, a stable and high-order numerical scheme is of importance for numerical simulations. To achieve stability, the total variation of a discrete solution in two consecutive time steps n and $n+1$ should be diminished $TV(\Phi^{n+1}) < TV(\Phi^n)$, so that numerical instability is avoided. A flux limiter function $\Psi(r)$ can be employed to constrain any possible overshoot of fluxes.

Using the TVD scheme, the quantities at the cell faces are formulated as:

$$\begin{aligned} \Phi_{\mathrm{w}} &= \Phi_{\mathrm{U}} + \frac{1}{2}\Psi(r_{\mathrm{w}})(\Phi_{\mathrm{C}} - \Phi_{\mathrm{U}}) \\ \Phi_{\mathrm{e}} &= \Phi_{\mathrm{C}} + \frac{1}{2}\Psi(r_{\mathrm{e}})(\Phi_{\mathrm{D}} - \Phi_{\mathrm{C}}) \end{aligned} \tag{3.17}$$

where

$$\begin{aligned} r_{\mathrm{w}} &= \left(\frac{\Phi_{\mathrm{U}} - \Phi_{\mathrm{UU}}}{\Phi_{\mathrm{C}} - \Phi_{\mathrm{U}}}\right) \\ r_{\mathrm{e}} &= \left(\frac{\Phi_{\mathrm{C}} - \Phi_{\mathrm{U}}}{\Phi_{\mathrm{D}} - \Phi_{\mathrm{C}}}\right) \end{aligned} \tag{3.18}$$

A collection of popular flux limiter functions $\Psi(r)$ is represented in Table 3.1 [119].

Function name	Limiter function $\Psi(r)$
CHARM [120]	$\Psi(r)$=$r(3r+1)/(r+1)^2$ if $r>0$ and $\Psi(r)$ if $r\leq 0$
Van Leer [121]	$\Psi(r)$=$(r+\lvert r\rvert)/(1+\lvert r\rvert)$
Van Albada 1 [122]	$\Psi(r)$=$(r+r^2)/(1+r^2)$
Van Albada 2 [123]	$\Psi(r)$=$2r/(1+r^2)$
SUPERBEE [124]	$\Psi(r)$=max[0,min(2r,1),min(r,2)]
Sweby [125]	$\Psi(r)$=max[0,min(βr,1),min(r,β)], $1<\beta<2$
UMIST [126]	$\Psi(r)$=max[0,min(2r, (0.25 + 0.75r), (0.75 + 0.25r), 2)]
Osher [127]	$\Psi(r)$=max[0,min(r,β)], $(1\leq\beta\leq 2)$
Koren [128]	$\Psi(r)$=max[0,min(2r, 0.5(1 + r), 2)], $(1\leq\beta\leq 2)$
Min-Mod [124]	$\Psi(r)$=min(r,1) if $r>0$ and $\Psi(r)=0$ if $r\leq 0$

TABLE 3.1: Limiter functions for the TVD scheme.

3.3 Temporal discretization

For a time-dependent partial differential equations, both spatial and temporal discretization are required. A general time-dependent equation can be expressed as follows:

$$\frac{\partial \Phi}{\partial t} = F(\Phi) \ . \tag{3.19}$$

The first order temporal discretization is formulated using backward differencing as:

$$\frac{\Phi^{n+1} - \Phi^{n}}{\Delta t} = F(\Phi) \ , \tag{3.20}$$

and a 2$^{\text{nd}}$ order formulation of the temporal discretization is outlined as follows:

$$\frac{3\Phi^{n+1} - 4\Phi^{n} + \Phi^{n-1}}{2\Delta t} = F(\Phi) \ . \tag{3.21}$$

In Eqs. 3.19-3.21, Φ^{n-1}, Φ^{n} and Φ^{n+1} denote a scalar quantity at the corresponding time steps $t - \Delta t$, Δt and $t + \Delta t$, consecutively.

3.3.1 Explicit time integration

In an explicit time integration, the scalar quantity Φ^{n+1} is determined from the RHS of the Eq. 3.19 at the previous time step n as:

$$\frac{\Phi^{n+1} - \Phi^{n}}{\Delta t} = F(\Phi^{n}) \tag{3.22}$$

The calculation using an explicit approach is simple and straightforward since $F(\Phi^{n})$ is already known. However, this approach is conditionally stable, in which the time step size Δt has to be restricted by the Courant-Friedirch-Lewy number (CFL number) [129] and sufficient diffusion is needed.

Using a Runge-Kutta method [130], higher order discretizations for the explicit time integration are achieved. The formulation for the 2$^{\text{nd}}$ order time discretization reads:

$$\begin{aligned} \Phi^{n+1/2} &= \Phi^{n} + \frac{\Delta t}{2} F(\Phi^{n}) \\ \Phi^{n+1} &= \Phi^{n} + \Delta t F(\Phi^{n+1/2}) \end{aligned} \tag{3.23}$$

The 4^{th} order formula for time integration can be employed to improve further the accuracy of the numerical results:

$$\begin{aligned}
\Phi^{n+1} &= \Phi^n + \Delta t\left(\frac{\kappa_1}{6} + \frac{\kappa_2}{3} + \frac{\kappa_3}{3} + \frac{\kappa_4}{6}\right) \\
\kappa_1 &= F(\Phi^n) \\
\kappa_2 &= F(\Phi^n + \kappa_1\frac{\Delta t}{2}) \\
\kappa_3 &= F(\Phi^n + \kappa_2\frac{\Delta t}{2}) \\
\kappa_4 &= F(\Phi^n + \kappa_3\Delta t)
\end{aligned} \tag{3.24}$$

3.3.2 Implicit time integration

In an implicit time integration, the scalar quantity Φ^{n+1} is evaluated by the RHS of the equation at the same time $n+1$. Therefore, the integration of Φ over time step Δt reads:

$$\Phi^{n+1} = \Phi^n + \Delta t F(\Phi^{n+1}) \tag{3.25}$$

Equation 3.25 can be solved by an iterative method. The implicit approach is unconditionally stable and can be used with large time steps.

To improve the accuracy of the time integration, one can apply the second-order Crank-Nicolson scheme [131] as:

$$\Phi^{n+1} = \Phi^n + \frac{1}{2}\Delta t\left(F(\Phi^{n+1}) + F(\Phi^n)\right) \tag{3.26}$$

3.4 Non-reflecting boundary conditions for engine simulation

The characteristic boundary conditions for compressible flow have been intensively investigated in many studies such as Hedstrom [132], Thompson [133, 134], Giles [135], Poinsot et al. [136], Nicoud [137], Yoo et al. [138, 139] or Lodato et al. [140]. This section presents the derivation of non-reflecting boundary conditions for a hyperbolic system in the conservative form. In the end, some available characteristic boundary

conditions for Navier-Stokes equations are provided. In the context of engine simulation, the non-reflecting boundaries are applied to impose the flow conditions at the inlet and the exhaust ports of the simulated engine. This section closely follows the work of Thompson [133, 134] and the book of Poinsot et al. [71].

A simple form (no diffusion) of hyperbolic governing equations of different physical conserved quantities is shown in the following.

The conservation of mass:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \tag{3.27}$$

The conservation of energy:

$$\frac{\partial e}{\partial t} + \frac{\partial (e+p) u_i}{\partial x_i} = 0 \tag{3.28}$$

$$\rho e = \frac{1}{2}\rho u_k u_k + \rho e_s \,, \quad e_s = \int_{T_0}^{T} c_v(T)\, dT \tag{3.29}$$

The conservation of momentum:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} + \frac{\partial p}{\partial x_i} = 0 \tag{3.30}$$

The conservation of species:

$$\frac{\partial \rho Y_i}{\partial t} + \frac{\partial \rho Y_i u_i}{\partial x_i} = \dot{\omega} \tag{3.31}$$

Equations 3.27, 3.28, 3.30 and 3.31 are rewritten in the vector form of conservative variables $\tilde{U}$ and fluxes F^1, F^2 and F^3 .

$$\tilde{U} = \begin{pmatrix} \rho \\ e \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho Y_i \end{pmatrix} \,, \tag{3.32}$$

and the vector of fluxes

$$F^1 = \begin{pmatrix} \rho u_1 \\ (e+p)u_1 \\ \rho u_1 u_1 + p \\ \rho u_2 u_1 \\ \rho u_3 u_1 \\ \rho Y_i u_1 \end{pmatrix}, \quad F^2 = \begin{pmatrix} \rho u_2 \\ (e+p)u_2 \\ \rho u_2 u_1 \\ \rho u_2 u_2 + p \\ \rho u_3 u_2 \\ \rho Y_i u_2 \end{pmatrix}, \quad F^3 = \begin{pmatrix} \rho u_3 \\ (e+p)u_3 \\ \rho u_3 u_1 \\ \rho u_3 u_2 \\ \rho u_3 u_3 + p \\ \rho Y_i u_3 \end{pmatrix} . \tag{3.33}$$

The primitive variable vector is defined as:

$$U = \begin{pmatrix} \rho \\ e \\ u_1 \\ u_2 \\ u_3 \\ Y_i \end{pmatrix} . \tag{3.34}$$

The relation between the conservative variable vector $\tilde{U}$ and the primitive variable vector U is given as

$$\frac{\partial \tilde{U}}{\partial t} = \boldsymbol{P} \frac{\partial U}{\partial t} \ , \tag{3.35}$$

where the Jacobian matrix is obtained from:

$$\boldsymbol{P} = \frac{\partial \tilde{U}}{\partial U} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2}u_i^2 & 1/(\gamma-1) & \rho u_1 & \rho u_2 & \rho u_3 & 0 \\ u_1 & 0 & \rho & 0 & 0 & 0 \\ u_2 & 0 & 0 & \rho & 0 & 0 \\ u_3 & 0 & 0 & 0 & \rho & 0 \\ Y_i & 0 & 0 & 0 & 0 & \rho \end{pmatrix} \tag{3.36}$$

Similarly, the relation between the vector of fluxes F^i and the primitive variable vector U can be written as:

$$\frac{\partial F^i}{\partial x_i} = \boldsymbol{Q^i} \frac{\partial U}{\partial x_i} \tag{3.37}$$

where the elements of the matrix $\boldsymbol{Q^i}$ are:

$$q^i_{jk} = \frac{\partial F^i_j}{\partial x_k} \tag{3.38}$$

The primitive form of the governing equations 3.27, 3.30, 3.28 and 3.31 can be formulated for the density:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} + \rho \frac{\partial u_i}{\partial x_i} = 0 \ , \tag{3.39}$$

for the velocity:

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} + \frac{1}{\rho}\frac{\partial p}{\partial x_i} = 0 \ , \tag{3.40}$$

for the energy:

$$\frac{\partial p}{\partial t} + u_j \frac{\partial p}{\partial x_j} + \gamma p \left(\frac{\partial u_j}{\partial x_j}\right) = 0 \ , \tag{3.41}$$

and for the species:

$$\frac{\partial Y_i}{\partial t} + u_i \frac{\partial Y_i}{\partial x_i} + Y_i \frac{\partial u_i}{\partial x_i} = 0 \ . \tag{3.42}$$

Considering the boundary condition in x_1 direction, a compact form of Eqs. 3.39- 3.42 in x_1 direction reads:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}^1 \frac{\partial \boldsymbol{U}}{\partial x_1} = 0 \ , \tag{3.43}$$

where

$$\boldsymbol{A}^1 = \begin{pmatrix} u_1 & 0 & \rho & 0 & 0 & 0 \\ 0 & u_1 & \gamma p & 0 & 0 & 0 \\ 0 & 1/\rho & u_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & u_1 & 0 \\ 0 & 0 & Y_i & 0 & 0 & u_1 \end{pmatrix} \ . \tag{3.44}$$

The eigenvalues of A^1 are :

$$\lambda_1 = u_1 - c, \lambda_2 = \lambda_3 = \lambda_4 = u_1, \lambda_5 = u_1 + c, \lambda_{5+i} = u_1 \ , \tag{3.45}$$

where $c = \frac{\gamma p}{\rho}$ represents the speed of sound. The eigenvalues λ_1 and λ_5 denote the velocities of sound waves traveling in the negative and the positive directions of x_1. The eigenvalues λ_2, λ_3, λ_4 and λ_{5+i} are the advection velocities of entropy, u_2, u_3 and the species Y_i, respectively.

The eigenvectors are written as:

$$\begin{cases} \boldsymbol{l}_1^T = (0, 1, -\rho, 0, 0, 0) \\ \boldsymbol{l}_2^T = (c^2, -1, 0, 0, 0, 0) \\ \boldsymbol{l}_3^T = (0, 0, 0, 1, 0, 0) \\ \boldsymbol{l}_4^T = (0, 0, 0, 1, 0, 0) \\ \boldsymbol{l}_5^T = (0, 1, \rho c, 0, 0, 0) \\ \boldsymbol{l}_{5+i}^T = (0, 0, 0, 0, 0, 1) \end{cases} \tag{3.46}$$

From the eigenvectors in Eq. 3.46, the amplitudes of the characteristic waves entering and leaving the computational domain are calculated:

$$\begin{cases} L_1 = \lambda_1 \left(\frac{\partial p}{\partial x_1} - \rho c \frac{\partial u_1}{\partial x_1} \right) \\ L_2 = \lambda_2 \left(c^2 \frac{\partial \rho}{\partial x_1} - \frac{\partial p}{\partial x_1} \right) \\ L_3 = \lambda_3 \frac{\partial u_2}{\partial x_1} \\ L_4 = \lambda_4 \frac{\partial u_3}{\partial x_1} \\ L_5 = \lambda_5 \left(\frac{\partial p}{\partial x_1} + \rho c \frac{\partial u_1}{\partial x_1} \right) \\ L_{5+k} = \lambda_{5+k} \frac{\partial Y_k}{\partial x_1} \end{cases} \tag{3.47}$$

Figure 3.3 demonstrates the conventional directions of the incoming waves at the inlet and outgoing waves at the outlet of the computational domain.

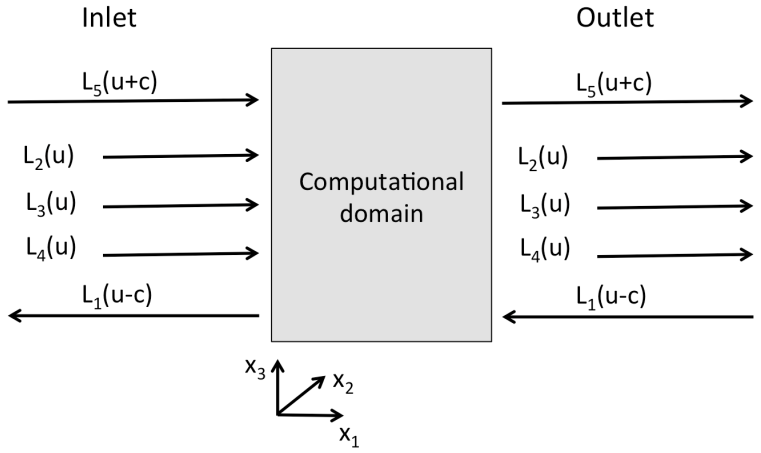

FIGURE 3.3: Characteristic waves entering and leaving of the domain.

The amplitudes of the incoming characteristic waves are computed from:

$$\begin{cases} L_1 = \lambda_1\big(\frac{\partial p}{\partial x_1} - \rho c \frac{\partial u_1}{\partial x_1}\big) \\ L_2 = \big(\gamma - 1\big) L_1 \\ L_3 = 0 \\ L_4 = 0 \\ L_5 = L_1 \end{cases} \tag{3.48}$$

where $\gamma = c_p/c_v$ is the isentropic exponent (ratio of the specific heats). In many cases, the acoustic waves could travel up-stream and interact with the inlet where the inflow velocity is imposed. This induces the reflection of the incident waves at the inlet section. To maintain the non-reflecting boundary conditions, one may compute the amplitudes of the incoming characteristic waves in the following way:

$$\begin{cases} L_2 = \sigma_2(T - T_{inlet}) \\ L_3 = \sigma_3(v - v_{inlet}) \\ L_4 = \sigma_4(w - w_{inlet}) \\ L_5 = \sigma_5(u - u_{inlet}) \end{cases} \tag{3.49}$$

In Eqs. 3.49, the value of the relaxation parameters σ_i is chosen in a certain range in oder to match the impedance of the inlet. However, σ_i has to be chosen carefully: with small or zero value of σ_i, the non-reflecting boundary is achieved at the inlet section but the target velocities are not maintained. On the other hand, maintaining the target inflow velocities requires larger values of σ_i, which leads to the reflection of the incident waves at the inlet section. By choosing reasonable values of σ_i, the mean values of the inlet velocities and the temperature are kept close to the target values, while at the same time the non-reflecting boundary is achieved. A more elaborate discussion and explanation can be found in the book of Poinsot et al. [71].

The amplitudes of the outgoing characteristic waves, e.g., at the exhaust port are computed from:

$$\begin{cases} L_1 = K\big(p - p_\infty\big) \\ L_2 = \lambda_2\big(c^2\frac{\partial \rho}{\partial x_1} - \frac{\partial p}{\partial x_1}\big) \\ L_3 = \lambda_3\frac{\partial u_2}{\partial x_1} \\ L_4 = \lambda_4\frac{\partial u_3}{\partial x_1} \\ L_5 = \lambda_5\big(\frac{\partial p}{\partial x_1} + \rho c\frac{\partial u_1}{\partial x_1}\big) \end{cases} \tag{3.50}$$

where $K = \sigma\big(1-M^2\big)c/\mathcal{L}$, M is the maximum Mach number in the flow, $\mathcal{L}$ is the domain length, σ is a chosen constant $(0, 1 < \sigma < \pi)$ and p_∞ is the static pressure at the far field.

Applying the obtained amplitudes of the waves to calculate the flow conditions at the boundary for the Navier-Stokes equations, one can rewrite the equations 3.27-3.28 in the form of a characteristic vector $\boldsymbol{d}$.

$$\boldsymbol{d} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_{5+k} \end{pmatrix} = \begin{pmatrix} \frac{1}{c^2}\left[L_2 + \frac{1}{2}(L_5 + L_1)\right] \\ \frac{1}{2}(L_5 + L_1) \\ \frac{1}{2\rho c}(L_5 - L_1) \\ L_3 \\ L_4 \\ d_{5+k} \end{pmatrix} = \begin{pmatrix} \frac{\rho u_1}{\partial x_1} \\ \rho c^2\frac{\partial u_1}{\partial x_1} + u_1\frac{\partial p_1}{\partial x_1} \\ u_1\frac{\partial u_1}{\partial x_1} + \frac{1}{\rho}\frac{\partial p_1}{\partial x_1} \\ u_1\frac{\partial u_2}{\partial x_1} \\ u_1\frac{\partial u_3}{\partial x_1} \\ u_1\frac{\partial Y_k}{\partial x_1} \end{pmatrix} \tag{3.51}$$

The boundary condition for the density ρ is:

$$\frac{\partial \rho}{\partial t} + d_1 + \frac{\partial \rho u_2}{\partial x_2} + \frac{\partial \rho u_3}{\partial x_3} = 0 \ . \tag{3.52}$$

The boundary condition for the momentum ρu_1 is:

$$\frac{\partial \rho u_1}{\partial t} + u_1 d_1 + \rho d_3 + \frac{\partial \rho u_1 u_2}{\partial x_2} + \frac{\partial \rho u_1 u_3}{\partial x_3} = \frac{\partial \tau_{1j}}{\partial x_j} \ . \tag{3.53}$$

The boundary condition for the momentum ρu_2 is:

$$\frac{\partial \rho u_2}{\partial t} + u_2 d_1 + \rho d_4 + \frac{\partial \rho u_2 u_2}{\partial x_2} + \frac{\partial \rho u_2 u_3}{\partial x_3} + \frac{\partial p}{\partial x_2} = \frac{\partial \tau_{2j}}{\partial x_j} \ . \tag{3.54}$$

The boundary condition for the momentum ρu_3 is:

$$\frac{\partial \rho u_3}{\partial t} + u_3 d_1 + \rho d_5 + \frac{\partial \rho u_3 u_2}{\partial x_2} + \frac{\partial \rho u_3 u_3}{\partial x_3} + \frac{\partial p}{\partial x_3} = \frac{\partial \tau_{3j}}{\partial x_j} \quad . \tag{3.55}$$

The boundary condition for the energy e is:

$$\begin{aligned} \frac{\partial e}{\partial t} + \frac{1}{2}\left(u_k u_k\right) d_1 + \frac{d_2}{\gamma - 1} + \rho u_1 d_3 + \rho u_2 d_4 + \rho u_3 d_5 + \frac{\partial}{\partial x_2}\left[(e+p)u_2\right] + \\ \frac{\partial}{\partial x_3}\left[(e+p)u_3\right] = \frac{\partial}{\partial x_i}(u_i \tau_{ij}) - \frac{\partial q_i}{\partial x_i} \quad . \end{aligned} \tag{3.56}$$

The boundary condition for the species ρY_k $(k = 1, .., N)$ is:

$$\frac{\partial \rho Y_i}{\partial t} + Y_k d_1 + \rho d_{5+k} + \frac{\partial}{\partial x_2}(\rho u_2 Y_k) + \frac{\partial}{\partial x_3}(\rho u_3 Y_k) = \frac{\partial}{\partial x_j}(M_{kj}) - \dot{\omega}_k \quad , \tag{3.57}$$

$$\text{where} \quad q_i = -\lambda \frac{\partial T}{\partial x_i}, M_{ki} = \rho D_k \frac{\partial Y_k}{\partial x_i} \quad . \tag{3.58}$$

It should be noted that, the amplitudes waves L_i and the vector $\boldsymbol{d}$ are computed differently depending on the inlet or outlet boundaries. Therefore, the equations 3.52-3.57 can generally be used to compute the flow conditions at the first ghost cells of the boundaries.

3.5 Mesh generation of engine geometry

Mesh generation is an important part of any numerical simulation, especially in the engine simulations since the accuracy of the numerical computations strongly depends on the quality of the mesh. In comparison to the complexity of conformal or non-uniform meshes, equidistant Cartesian grids are simpler and convenient to generate from the engine computer-aided design (CAD) models. In this work, we refer to this procedure as voxelization - a procedure for converting a conformal/unstructured mesh to an equidistant Cartesian mesh. Each cubic element of the Cartesian grid is referred to as a voxel. Figure 3.4 presents the computational grids of an engine geometry including the unstructured and the equidistant Cartesian grids.

It must be stressed that using Cartesian grids for a complex geometry of an internal combustion engine (with the intake and the exhaust manifold, cylinder head, intake and exhaust valves) requires a certain resolution to properly describe all the geometrical

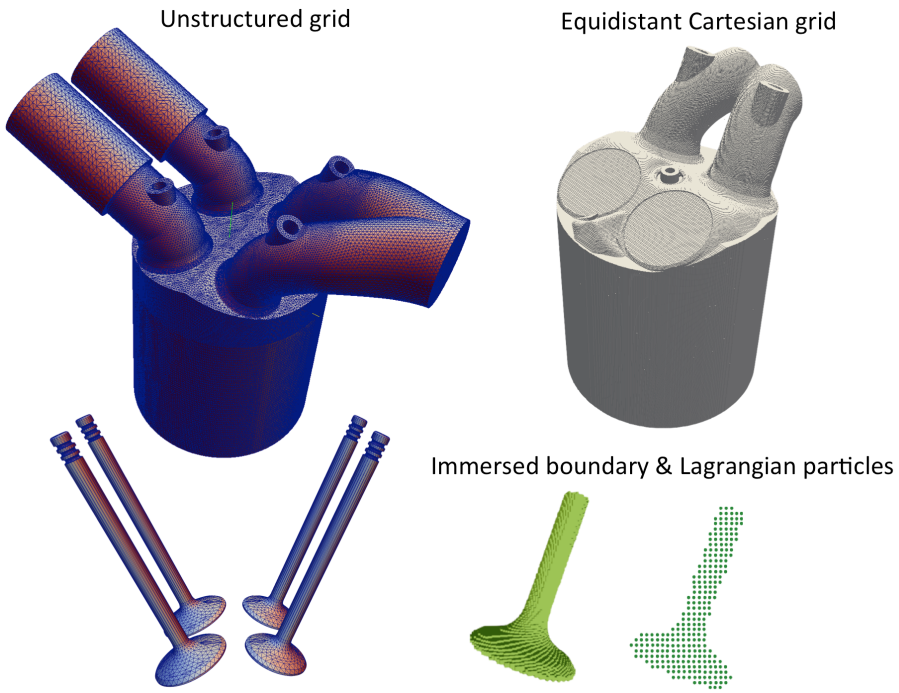

FIGURE 3.4: The unstructured and the Cartesian grids (Δ=0.2 mm) of the engine geometry and valves of the Darmstadt engine [141].

features of an engine geometry-like. Therefore, the equidistant Cartesian grid usually requires more computational cells than the unstructured grid does. Due to its simplicity, generation of an equidistant Cartesian grid for the engine simulation is much less complex than the work to create the unstructured meshes. However, there are several disadvantages of using this type of grid in the engine simulations. Readers are referred to Chapter 6 for a more elaborate discussion on this issue. Despite some drawbacks, equidistant Cartesian grids offer many advantages that make the approach as attractive as the body-fitted method (in the author opinion): simple in mesh generation, easy in programming, efficient in parallelization, and most suitable for LES due to the high accuracy, homogeneous, isotropic filter widths.

3.6 Lagrangian particles and moving boundary

Handling the moving boundaries in the body-fitted methods requires mesh-generation before or during the simulation. These approaches become even more challenging when the motion of the moving boundaries is not known in advance. Therefore, dynamic mesh generation is required, which is generally sophisticated and computationally expensive. Moreover, in the approaches using dynamic mesh generation, transferring data

from the old mesh to the new mesh requires some mapping procedures. Obviously, the interpolation and extrapolation in the mapping operations make the method more dissipative and inaccurate. An alternative approach is to use Lagrangian particles and immersed boundary method for the description of the moving objects. In this approach, the moving boundaries are marked by Lagrangian particles and the transportation of these particles represents the motion of the boundaries. Figure 3.5 presents the comparison between the two approaches, in which the moving valve is handled by the dynamic mesh generation and by the transport of Lagrangian particles.

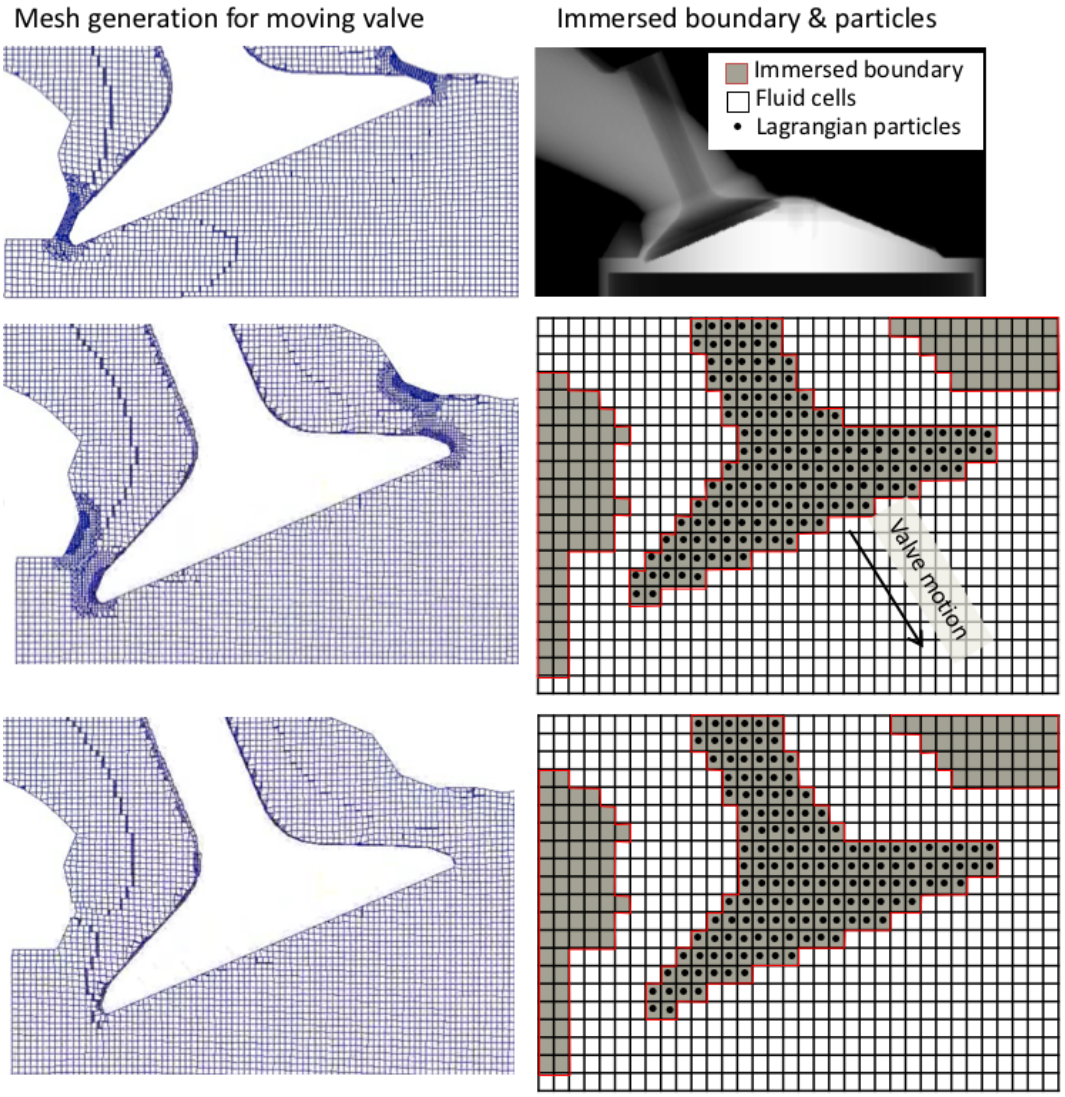

FIGURE 3.5: Moving boundary method by body-fitted approach [142] vs the combination of Lagrangian particles and immersed boundaries.

Generation of Lagrangian particles

There are two flexible ways of using Lagrangian particles for the numerical description

of the moving boundaries. Either one can use a single particle or many particles per computational cell, as is illustrated in Fig. 3.6. Using a single particle per cell is simpler and suitable for the problems where the motion of the objects follows the conventional directions. For the problems involving the rotation of the moving boundaries, using multiple particles per cell is more suitable to preserve the geometry since there are no change in the distances between the particles after rotation but the corresponding immersed boundaries may not be well-preserved. However, the numerical treatments of these methods are similar. In case of more particles in one cell, the averaging of the data such as velocity or acceleration of the particles in one cell will be carried out. In the scope of this thesis, only single particle per cell is used and the validations of different test-cases are provided.

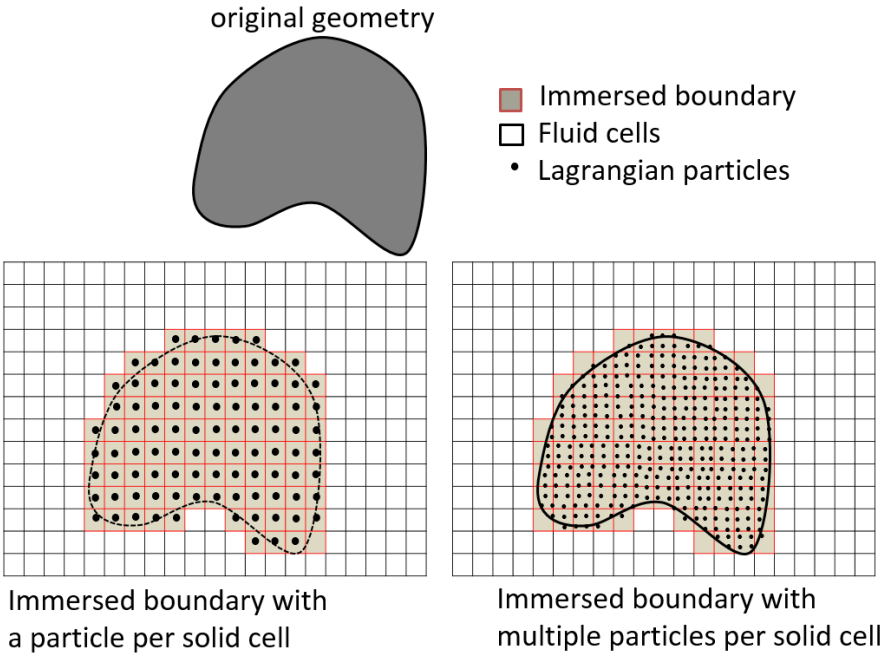

FIGURE 3.6: Geometric description using single and multiple particles per solid cell.

Since the particles need to be generated from the geometric objects prior to the simulation, transforming the moving objects to Lagrangian particles must be carried out. There are different ways to perform such task. The general steps are explained in the procedure 1 together with Fig. 3.7. In order to convert the moving boundaries as well as the stationary boundaries to a set of voxels, the voxelization step can be performed using available software such as Binvox [143].

Procedure 1: Generating Lagrangian particles for moving geometries

- Perform the voxelization of the original geometries with the same resolution Δ (single particle approach) of the computational domain
- Map the generated voxels of the moving boundaries onto the main computational grid for correct positioning
- Assign the indices and physical positions of the mapped voxels to the corresponding logic particles
- Write the information of all the logic particles including the indices and the physical positions that can be read later into the simulation

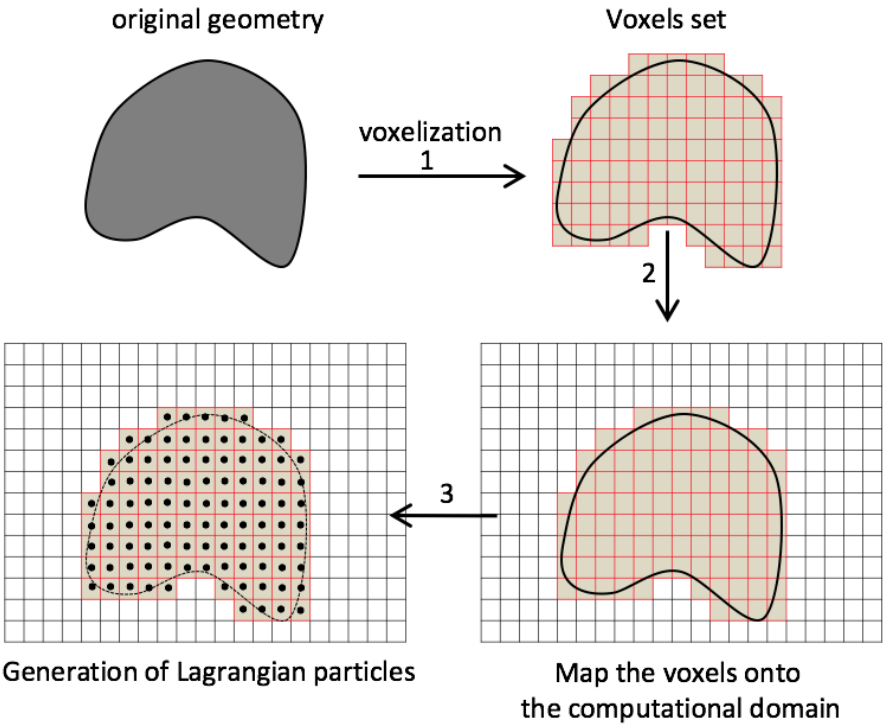

FIGURE 3.7: Three-step procedure to generate Lagrangian particles from the original geometry. The voxel size must be the same as the cell size Δ of the computational domain.

Moving boundary using Lagrangian particles

Moving a boundary, which is numerically described by Lagrangian particles, can be performed by particle transport. The equation for the transportation of particles in

time and space is outlined as:

$$x_i^n = x_i^{n-1} + v_{pi}^{n-1}\Delta t \ , \tag{3.59}$$

where x_i^n, x_i^{n-1} and v_{pi}^{n-1} denote the locations and the velocity of the particle i at the time steps n and $n-1$, respectively.

In many cases, the motion of the moving boundaries including the piston and valves are prescribed in time and space. For instance, in a 4-stroke engine, the motion of the piston, the intake and the exhaust valves are controlled and synchronized by the rotation of the crankshaft. Therefore, instead of solving the Eq. 3.59 for the new position of the particle, the time dependent velocity $v_{pi}(t)$ and the acceleration of the particle $a_{pi}(t)$ are determined from the prescribed positions in time. The velocity $v_{pi}(t)$ and the acceleration $a_{pi}(t)$ are used to impose the flow conditions at the moving boundaries:

$$v_{pi}^n = \frac{x_i^{n+1} - x_i^n}{\Delta t} \tag{3.60}$$

$$a_{pi}^n = \frac{v_{pi}^{n+1} - v_{pi}^n}{\Delta t} \tag{3.61}$$

In the context of the engine simulation, the velocity and the acceleration of the piston can be analytically computed from the piston kinematics using the crank angle (CA) $\Theta = \Theta(t)$, the conrod length l, the crank radius r, and the angular velocity ω ($\omega = 2\pi\mathrm{rpm}/60$) from the following Eqs. 3.62-3.63 [106]:

$$v_{pi} = \left[-r\sin\Theta - \frac{r^2\sin\Theta\cos\Theta}{\sqrt{l^2 - r^2\sin^2\Theta}} \right]\omega \tag{3.62}$$

$$a_{pi} = \left[-r\cos\Theta - \frac{r^2(\cos^2\Theta - \sin^2\Theta)}{\sqrt{l^2 - r^2\sin^2\Theta}} - \frac{r^4(\cos^2\Theta\sin^2\Theta)}{\sqrt{l^2 - r^2\sin^2\Theta}} \right]\omega \tag{3.63}$$

After updating the actual particle positions x_i^n, the logical particle coordinates (i,j,k) within the equidistant Cartesian grid must also be recalculated:

$$i = \left\lfloor \frac{x_i}{\Delta} \right\rfloor + 1 + \mathrm{nG} \ , \tag{3.64}$$

where nG is the number of ghost cells of the computational grid (usually $\mathrm{nG} = 2$) and Δ is the cell size. This calculation is used to indicate whether the particle already moves

to the new cell with new indices i,j and k. After that the corresponding immersed boundary can be updated accordingly.

Since the particles are already moved to the new position, the new interface between the fluid and the solid zone must be established. This can be done by blocking the cells of an immersed boundary field (IBF) where the particles are located. The indices i, j, k of a particle are also its position in the 3D array of the IBF.

Imposing the flow conditions on the moving interface

There are several approaches to determine the effect of the moving boundaries on the surrounding fluid. One popular method is to introduce a forcing function $\boldsymbol{f_b}$ and then modify the equation system $\mathrm{L}(\boldsymbol{\Phi}) = 0$ to the new one $\mathrm{L}(\boldsymbol{\Phi}) = \boldsymbol{f_b}$, where $\boldsymbol{\Phi}$ is the vector of conservative variables. In the continuous forcing approach, the function $\boldsymbol{f_b}$ is included before the discretization of $\mathrm{L}(\boldsymbol{\Phi}) = \boldsymbol{f_b}$. In the discrete forcing approach, $\boldsymbol{f_b}$ is added after the discretization of the system of equations [144]. The determination of the forcing function for the flow around a rigid body requires an iterative calculation.

In this work, the effect of the moving boundaries Γ_b on the surrounding fluid is determined via a new technique that employs the 'mirror flow' concept [145] in combination with the immersed boundaries and the Lagrangian particles. In this technique, the 'mirror flow' quantities $\boldsymbol{\Phi_\Gamma}$ of the solid cells at the moving boundaries are computed and imposed before solving the the discretized governing equations $\mathrm{L}(\boldsymbol{\Phi}) = 0$ in the next time step t_{n+1}, as illustrated in the procedure 2. To be precise, the explicit time-stepping from the time step t_{n+1} is performed for the moving boundary before advancing the whole numerical solution to the time step t_{n+1}. In order to apply the proposed method to engine simulation, detailed implementation steps are presented for the motored- and fired-engine in the procedure 3.

Procedure 2: Imposing the flow conditions at the moving boundaries

- Solve the discretized Navier-Stokes equations $\mathrm{L}(\mathbf{\Phi}) = \mathbf{0}$ for $\mathbf{\Phi^n}$
- Calculate the 'mirror flow' quantities $\mathbf{\Phi_\Gamma}$ of the solid cells inside the moving boundaries Γ_b using the predetermined particle's properties v_{pi}^{n+1}, a_{pi}^{n+1} and the extrapolated quantites from the neighbouring fluid cells $\mathbf{\Phi^n}$ of the particle
- Solve the discretized Navier-Stokes equations $\mathrm{L}(\mathbf{\Phi}) = \mathbf{0}$ for $\mathbf{\Phi^{n+1}}$

The velocity v_{si}^{n+1} and the pressure gradient $\partial p^{n+1}/\partial x_i$ of the solid cells at the solid-fluid interfaces at time step t_{n+1} are calculated according to Forrer et al. [145]:

$$v_{si}^{n+1} = 2v_{pi}^{n+1} - \frac{\sum_{j=1}^{N_{nb}} v_{fij}^{n}}{N_{nb}} \quad , \tag{3.65}$$

$$\frac{\partial p^{n+1}}{\partial x_i} = -\frac{\sum_{j=1}^{N_{nb}} \rho_j^n}{N_{nb}} a_{pi}^{n+1} \quad . \tag{3.66}$$

Here, v_{pi}^{n+1} and a_{pi}^{n+1} denote the velocity and the acceleration of the particle i at the time step t_{n+1}. In equations 3.65-3.66, the averaged quantities of the neighbouring fluid cells N_{nb} $(0 \leq N_{nb} \leq 6)$ $\sum_{j=1}^{N_{nb}} v_{fij}^n/N_{nb}$ and $\sum_{j=1}^{N_{nb}} \rho_j^n/N_{nb}$ at the time step t_n are required to compute the the numerical quantities of a solid cells.

The energy e_{si} of the moving wall can be calculated from the temperature T_{si} and the velocity v_{si} of the solid cells:

$$e_{si} = \frac{1}{2} v_{si}^2 + c_v T_{si} \quad . \tag{3.67}$$

The heat flux at the moving walls $q_i = -\lambda \frac{\partial T}{\partial x_i}$ is also considered in the transport of energy, in which the isothermal boundary condition is applied with a fixed wall temperature. The temperature at the moving and the stationary walls is set to the same wall temperature T_{wall}. No additional modeling of the wall heat transfer is used in the current work.

Procedure 3: Moving boundaries in engine simulation-implementation

- Compute the pressure using the ideal gas law.
- Calculate the convective- and diffusive fluxes of conserved quantities at cell faces.
- Update the conserved quantities with the convective and diffusive fluxes.
- Calculate the pressure gradient and update the momentum.
- Calculate the source term of the FSD equation and update the progress variable (this step applies to the fired-engine).
- Solve the ODE for the Lagrangian particles (e.g., the groups of particles belong to the moving valves and moving piston).
- Block the cells where the particles are located.
- Calculate and impose the non-reflecting boundary conditions at the inlet and outlet of the engine.
- Impose the wall-bounded conditions of the physical quantities for the boundary cells of the IB.
- Calculate and update the momentum at the moving boundaries.
- Update the velocity at the cell- center and faces.
- Update the energy with the RHS.
- Impose the wall-bounded condition for the energy.
- Calculate and update the sub-grid viscosity.

Chapter 4

Test cases

In the scope of this work, the numerical treatments for the moving boundaries are based on the combination of the immersed boundary method (IBM) and the Lagrangian particle approach. Therefore, it is important to examine the suitability of the IBM in numerical simulations of the fluid flow. In this chapter, the verification of the IBM in the flow simulation is carried out for three different test cases: (1) prediction of the drag coefficient of a cylinder in a fluid flow; (2) prediction of the velocity profiles and the velocity fluctuations in a horizontal- and an inclined channel; (3) prediction of the flow velocity in a simplified engine geometry with a moving piston and a fixed central valve.

4.1 Test case 1: flow around a cylinder

In this test case, the IBM was validated by predicting the drag coefficient of an immersed cylinder in the flow field at different Reynolds numbers, namely 10000, 25000 and 50000. Since the drag coefficient strongly depends on the geometry, a good agreement for the prediction of the drag coefficient between the simulated results and the measurements would indicate the suitability of IBM for flow simulations. The configuration of this test case is presented in Table 4.1 and Fig. 4.1.

In this test case, the pressure distribution method [146] is used to calculate the drag coefficient C_d from the pressure coefficient C_p as:

$$C_p = \frac{p - p_\infty}{0.5\rho U_\infty^2} \quad , \tag{4.1}$$

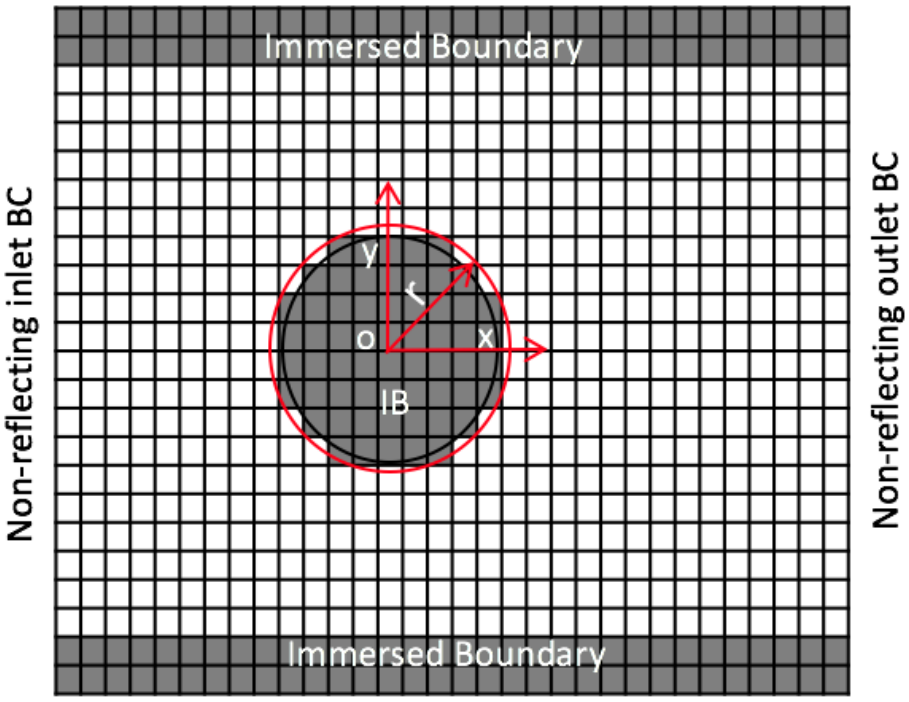

FIGURE 4.1: Configuration of the test case 1: flow around a cylinder which is described by IBM. Non-reflecting boundary conditions are applied at the inlet and at the outlet of the domain. The drag coefficient is evaluated on the circumference of the cylinder (illustrated by the outer circle with the radius r).

where p and ρ are the local pressure and the local density, respectively, p_∞ and U_∞ denote the inlet pressure and the inlet velocity at the far field of the domain. Figure 4.2 shows the average value of the pressure coefficient over time for the test case with Reynolds number of 25000. According to the pressure distribution method, the drag coefficient C_d is evaluated by integrating the pressure coefficient C_p over the cylinder circumference (from 0 to 2π). It is easy to see that the flow is symmetric in axial direction (see Fig. 4.2). Therefore, one just has to integrate C_p over a half of the circumference (from 0 to π) to obtain C_d as:

$$C_d = \int_0^\pi C_p \cos\theta \, \mathrm{d}\theta \approx \sum_{i=1}^{N} C_{pi} \cos\theta_i \Delta\theta \quad . \tag{4.2}$$

Here, half of the circumference of the cylinder is equally divided into $N-1$ segments by an angle of $\Delta\theta$. C_{pi} represents the local pressure coefficient at angle θ_i $(0 \leq \theta_i \leq \pi)$ on the circumference. Figures 4.1 and 4.2 illustrate a circle where C_{pi} of each point is interpolated from the fluid neighbouring cells. It must be noted that the IBM, unlike the body-fitted methods, cannot numerically describe the exact data points on the surface

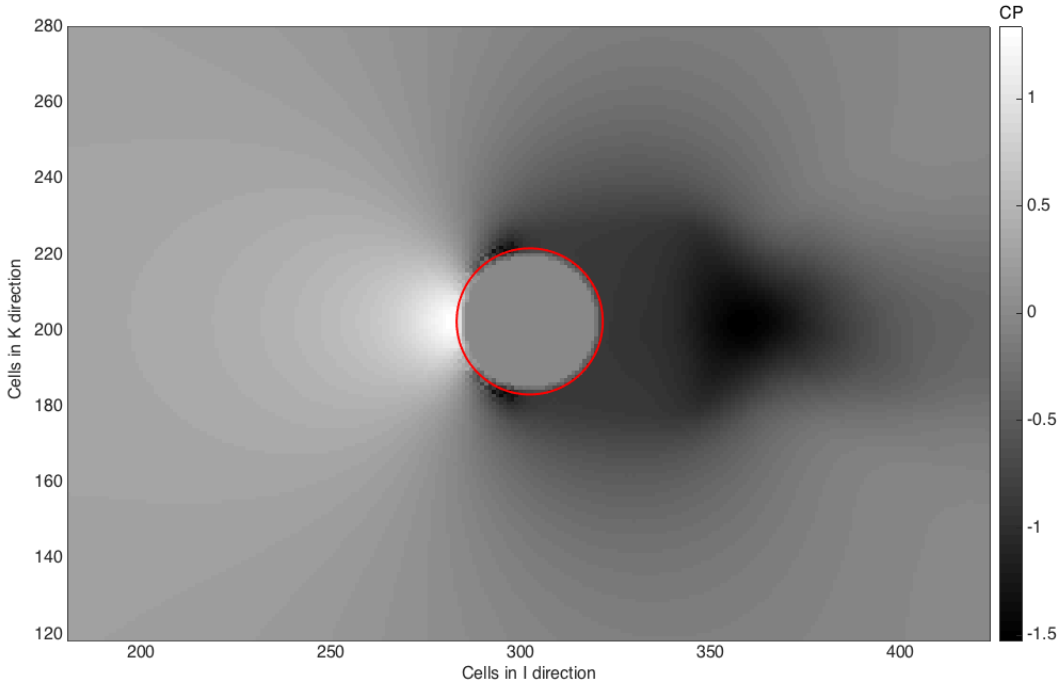

FIGURE 4.2: Pressure coefficient field close to the immersed cylinder. The data points on the cylinder surface are interpolated and integrated to calculate the drag coefficient of the cylinder.

of the cylinder, which makes an interpolation necessary.

Re	CPUs used	Domain size	Cell size Δ	Cylinder radius R
10000	48	604×44×404	0.5 mm	19.25 mm
25000	48	604×44×404	0.5 mm	19.25 mm
50000	48	604×44×404	0.5 mm	19.25 mm

TABLE 4.1: Configuration for the simulations of the flow around a cylinder.

Table 4.2 shows the drag coefficients as obtained by the simulation in comparison to the experiments [147]. A reasonable agreement between the simulations and the experiments is achieved. In the case of Re = 10000, a large discrepancy between the simulation and the measurement may be due to a very small velocity of the fluid, which leads to severe deficiencies in both efficiency and accuracy of the density-based solver for a compressible flow.

It should be noted that the numerical prediction of the drag coefficient is very sensitive to the geometrical features and the grid resolution near the wall [148, 149, 150]. In the case of a flow around the cylinder, the calculation strongly depends on how well the curvature of the cylinder is described by an IBM; in this work by an equidistant Cartesian grid.

Re	Sim	Exp	Error
10000	1.26	1.11	13.5 %
25000	1.22	1.16	4.6 %
50000	1.15	1.20	4.3 %

TABLE 4.2: A comparison between the simulated and measured drag coefficients of an immersed cylinder.

Using IBM for a flow simulation around the cylinder, Lai et al. [151] have demonstrated that, when the grid resolution increases by a factor of 2, the discrepancy between the experimental data and the simulation results reduces by a factor of 2, i.e. a first order convergence rate is achieved. The test case has shown that, if the grid resolution is fine enough, the IBM can be efficiently used to describe complex geometries in the flow simulation.

4.2 Test case 2: LES of a channel flow

To further examine the suitability of IBM for LES, channel flow simulations were performed with two different geometrical configurations including normal- and inclined channels, as illustrated in Figs. 4.3 and 4.5. In the normal-channel, the computational grid aligns with the flow direction and the immersed wall is exactly described by the IBM, as shown in Fig. 4.3. The inclined-channel is designed to test the IBM in unfavorable conditions where the flow direction is inclined at an angle of 45° to the grid orientation and the channel walls are not exactly described by the IBM, as depicted in Fig. 4.5. As this situation is common in engine simulations using IBM, it is necessary to examine the quality of the numerical predictions in these settings.

The normal- and the inclined channels in these test cases are set up as an infinite length channel discretized by an equidistant Cartesian grid. Therefore, periodic boundary conditions for the conserved quantities are imposed at the inlet and the outlet of the channel. The flow in the channel is driven by a pressure gradient $\partial p/\partial x$, which is calculated from the turbulent Reynolds number Re_τ, the kinematic viscosity ν, the fluid

density ρ, the friction velocity u_τ, the wall shear stress Γ_τ and the channel height h, as:

$$\begin{aligned} \frac{\partial p}{\partial x} &= \frac{\Gamma_\tau}{0.5h} \ , \\ \Gamma_\tau &= u_\tau^2 \rho \ , \\ u_\tau &= \frac{\nu \mathrm{Re}_\tau}{0.5h} \ . \end{aligned} \tag{4.3}$$

In simulations of the channel flows, capturing the boundary layer flow properly is essential, since it has a strong influence on the external flow. In common approaches to resolve the boundary layer, local refinement near the wall is required and the first grid point off the wall should be in the laminar sub-layer region ($y^+ < 5$). For the simulations using an equidistant Cartesian grid without the local refinement, wall functions can be used as an alternative way to estimate the friction velocity or the turbulent viscosity at the first grid point off the wall. In this test case, two wall models proposed by Piomelli et al. [152] and by Wang et al. [153] were employed to assess their impact on the numerical predictions.

The obtained simulation results including the mean velocity profile and RMS of the three velocity components are compared against the DNS data by Moser et al. [154] for $\mathrm{Re}_\tau = 395$. The results are presented in sections 4.2.1 and 4.2.2.

4.2.1 Simulations of a normal channel ($\mathrm{Re}_\tau = 395$)

The computational grid of the normal channel is illustrated in Fig. 4.3. Three simulations of the channel flow were performed with and without the wall models. The turbulent viscosity ν_τ was evaluated using the Sigma model [87] with the model constant $C_\sigma = 1.5$. In the case with the wall models, ν_τ of the first grid point off the wall was calculated from the wall models (see Fig. 4.3).

Test case	CPUs used	Domain size	Subgrid-model	Wall model
1	48	304×94×154	Sigma model, C_σ=1.5	-
2	48	304×94×154	Sigma model, C_σ=1.5	Piomelli et al. [152]
3	48	304×94×154	Sigma model, C_σ=1.5	Wang et al. [153]

TABLE 4.3: Numerical configurations for the simulations of the normal channel.

Detailed comparisons between the LES results of the normal channel and the DNS data are presented in Figure 4.4. Clearly, the simulation using the wall model by Piomelli et

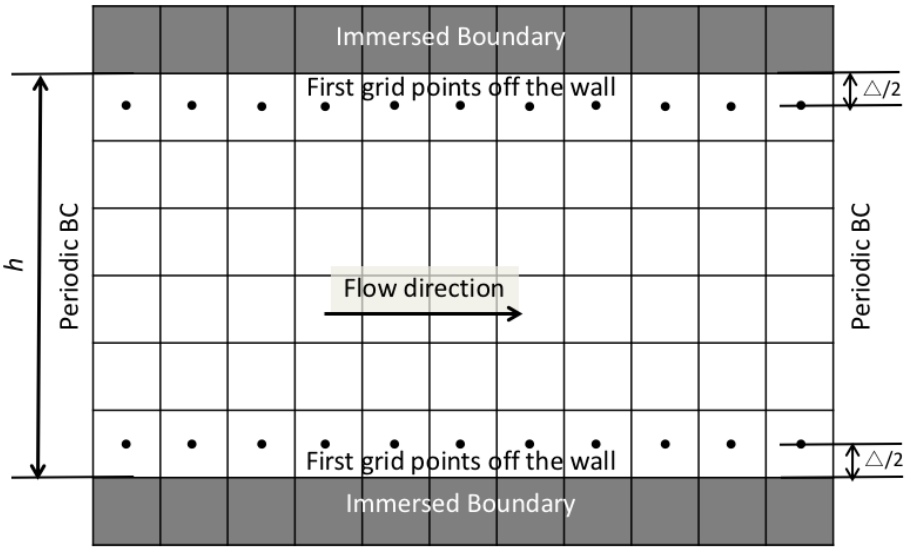

FIGURE 4.3: Configuration of the normal channel where the flow is perpendicular to the cells.

al. [152] yields the best agreement with the DNS data for both the velocity profile and the RMS. In the simulation using the dynamic wall model by Wang et al. [153], the velocity profile shows a slight improvement in comparison to the results without wall model. The RMS of the simulation without the wall model shows a better agreement with the DNS data. It can be seen in Fig. 4.4 that the case without the wall model yielded the largest value of the ratio of the turbulent to the laminar viscosity for the first grid point off the wall. It is possible that the overprediction of the turbulent viscosity ν_τ close to the wall may lead to the discrepancy between the LES and the DNS data in the middle of the channel.

It must be stressed that the wall model by Piomelli et al. [152] is only suitable for cases with a fully developed boundary layer as the turbulent Reynolds number is known. The model by Wang et al. [153], on the other hand, employs the numerical results that were obtained from the previous time step to compute the viscosity at the first grid point off the wall, which is more practical for the engine simulations. Together a good agreement between the numerical results from LES using IBM and DNS confirms the suitability of the IBM in the flow simulations.

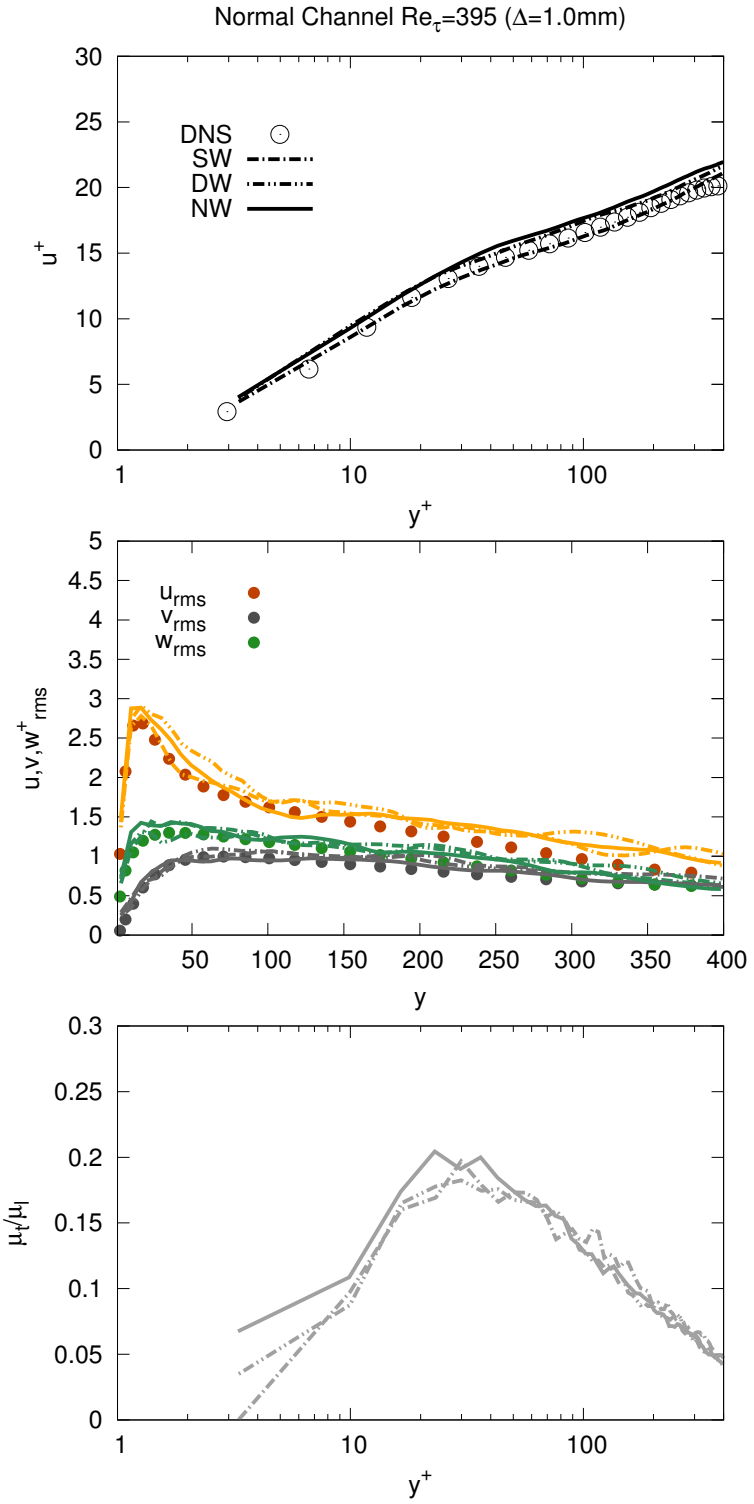

FIGURE 4.4: Comparison between DNS and LES results for the velocity profile, RMS and the ratio of the turbulent to the laminar viscosity. SW: LES with the wall model by Piomelli et al. [152], DW: LES with the dynamic wall model by Wang et al. [153], NW: LES without the wall model.

4.2.2 Simulations of 45° inclined channel ($Re_\tau = 395$)

In order to examine the IBM in more complex geometries, the numerical simulations of the fluid flow in an inclined channel were performed. In this setup, the misalignment between the flow direction and the grid orientation was introduced to illustrate the common situations in engine simulations using IBM. Figure 4.5 presents a 45° inclined channel discretized by an equidistant Cartesian grid, which is set up as an infinite length channel by using the periodic boundary conditions. Flow data are exchanged between the grid cells (denoted by the same color in Fig. 4.5) at the inlet and the outlet of the channel. Since the inclined walls were not accurately described by the IBM and the flow direction was not aligned with the grid orientation, achieving a good numerical prediction with this test case is more challenging than the simulations of the normal channel are. Therefore, besides the test cases with the wall models, the Germano procedure [89] for the dynamic model constant C_σ (Sigma model [87]) was also employed in the test cases to check for a possible improvement in the numerical results. The numerical setup of all the test cases for the simulations of the inclined channel is given in Table 4.4.

Test case	CPUs used	Domain size	Cell size	Subgrid-model	Wall model
1	192	180×64×180	1.0 mm	Sigma model, C_σ=1.5	-
2	192	180×64×180	1.0 mm	Sigma model, dynamic proc	-
3	192	180×64×180	1.0 mm	Sigma model, C_σ=1.5	Piomelli et al. [152]
4	192	180×64×180	1.0 mm	Sigma model, C_σ=1.5	Wang et al. [153]
5	600	354×124×354	0.5 mm	Sigma model, C_σ=1.5	-
6	600	354×124×354	0.5 mm	Sigma model, dynamic proc	-
7	600	354×124×354	0.5 mm	Sigma model, C_σ=1.5	Piomelli et al. [152]
8	600	354×124×354	0.5 mm	Sigma model, C_σ=1.5	Wang et al. [153]

TABLE 4.4: Numerical configurations for the simulations of the inclined channel.

Figure 4.6 shows the instantaneous- and the mean axial velocity field for the test case NW-0.5 mm. Profiles of the time-averaged mean velocity, the corresponding RMS of the velocity and the ratio of the turbulent- to the laminar viscosity were evaluated along a chosen line which is perpendicular to the virtual wall (the dash lines in Fig. 4.5). As it can be seen in Fig. 4.7, the discrepancy between the obtained simulation results of the same grid resolution was quite small for all the test cases. No significant impact of the wall models in the numerical results was observed. In comparison to the DNS data, the simulation results obtained from the test case using the dynamic procedure [89] (GM-1.0 mm) showed an improvement in terms of the mean velocity. The RMS of the axial

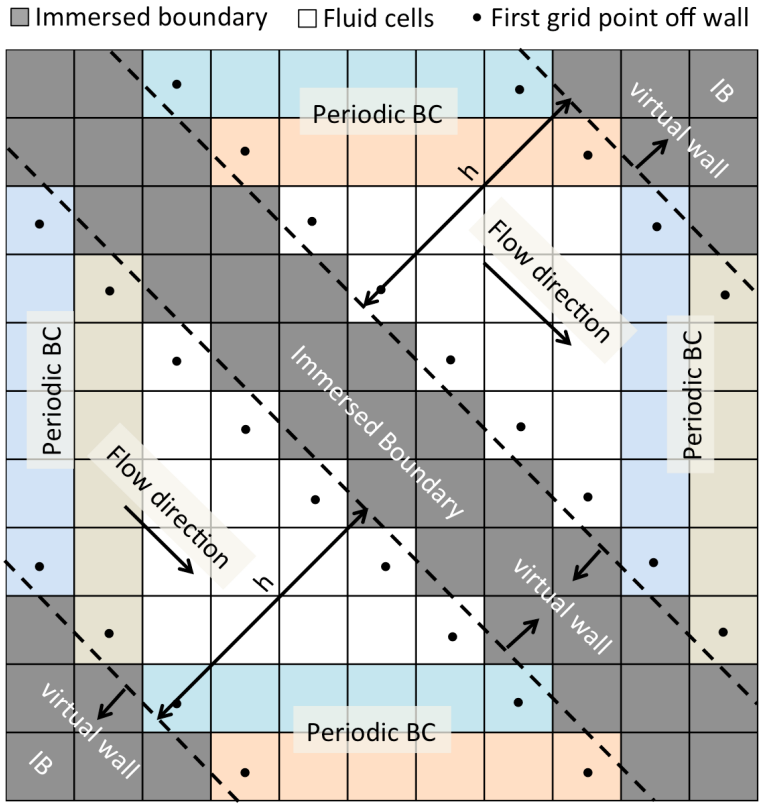

FIGURE 4.5: Configuration of an 45° inclined channel of height h. The distance between the wall and the first grid point off the wall is the distance between the dash line and the first grid point off the wall. Periodic boundary conditions are illustrated as the flow data are exchanged between the grid cells with the same colors.

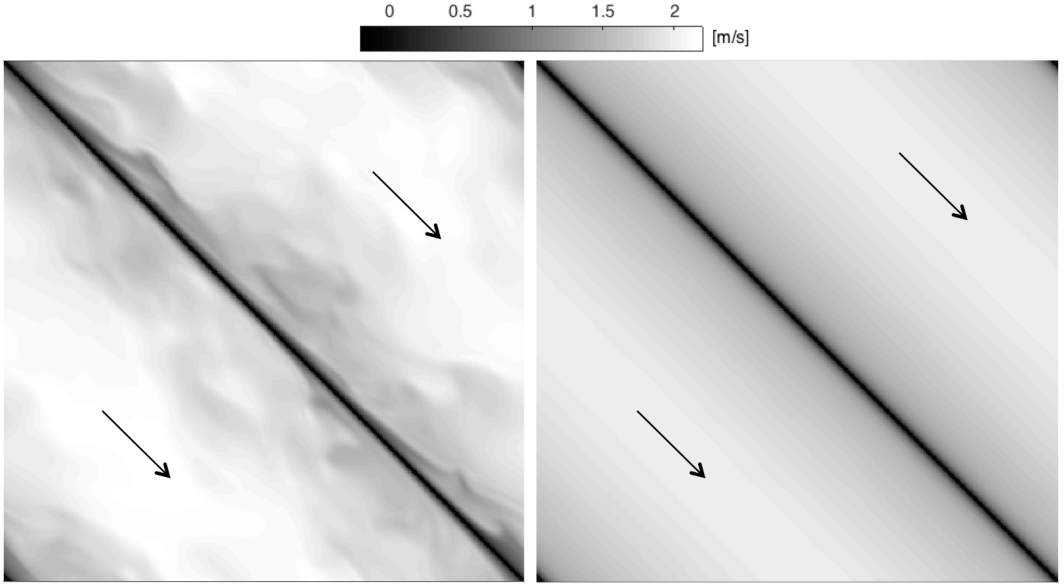

FIGURE 4.6: Axial velocity of test case NW-0.5mm of the inclined channel (⟶: flow direction; left: instantaneous velocity; right: mean velocity).

velocity u_{rms}, however, became less accurate despite the intensive computational effort spent on the dynamic procedure to locally calculate the value of the model constant C_σ.

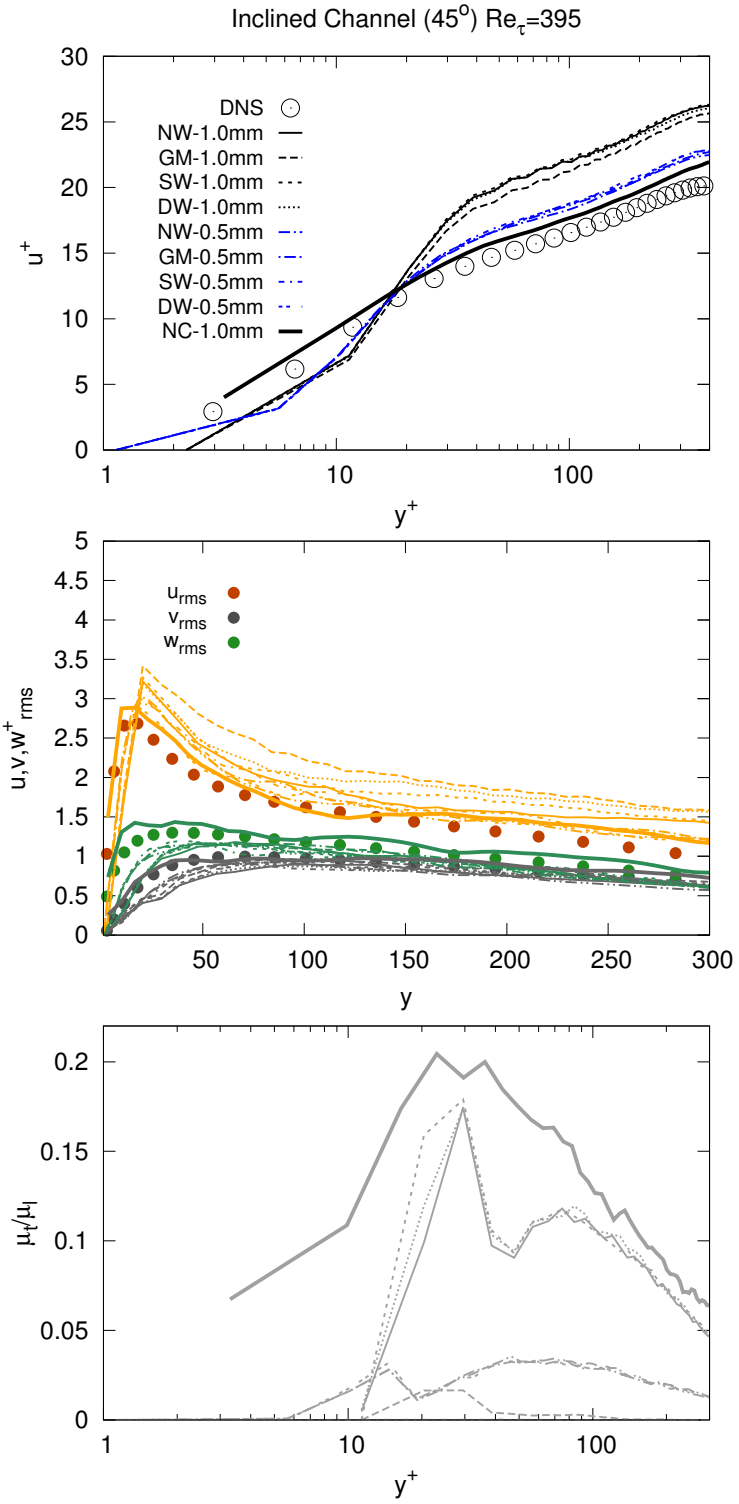

FIGURE 4.7: Comparison between DNS and LES results for the velocity profile, RMS and the ratio of the turbulent to the laminar viscosity. SW: LES with the wall model by Piomelli et al. [152], DW: LES with the dynamic wall model by Wang et al. [153], GM: LES with the dynamic procedure [89], NW: LES without the wall model, NC: LES of the normal channel without the wall model.

It is easy to see that close to the wall, the u^+ obtained by the LES simulations was much smaller than the u^+ of the DNS result for the same value of y^+. One possible explanation for the underprediction of u^+ at the first grid point off the wall is the geometrical feature of the computational grid near the walls. Considering the flow direction, each first grid point off the wall lies in between two solid triangles, that block the fluid motion (see Fig. 4.5), hence reducing the fluid velocity close to zero. The low value of the u^+ in the area near the wall would lead to the higher velocity in the middle region of the channel, to maintain the similar mass flow rate under the same channel configuration.

Comparing to the test cases of the inclined channel using the cell size of 1.0 mm (see Fig. 4.7), clear improvements in the velocity profile and the corresponding RMS were achieved with the refinement of the computational grid (Δ=0.5 mm). However, it is easy to see in Figs. 4.4 and 4.7 that a better agreement of DNS data for the velocity profile and the RMS was achieved by the simulations of the normal channel using the coarse grid (Δ=1.0 mm), in comparison to the simulation results from the fine grid (Δ=0.5 mm) of the inclined channel. Obviously, the misalignment between the flow direction and the grid orientation induced a strong impact on the simulation results. In general, the LES of the channel flow can still yield reasonably good results with a grid resolution fine enough for the description of the geometry, even though the inclined wall was not precisely represented by the IBM.

4.3 Test case 3: moving boundary in a simplified engine geometry

To demonstrate the application of the Lagrangian particles for the description of the moving boundary and to test the IBM in a more complex geometry, a multi-cycle engine simulation of a simplified engine geometry was performed. In this test case, the central valve was fixed and the fluid was sucked through the 4 mm-wide annular gap between the valve and the valve seat due to the pressure difference created by the motion of the piston. Detailed information of the engine geometry is illustrated in Fig. 4.8.

In comparison to the simulations of the inclined channel, this test case presented a more difficult situation where the misalignment between the fluid flow and the grid orientation was even more extreme due to the complexity of the engine geometry, as

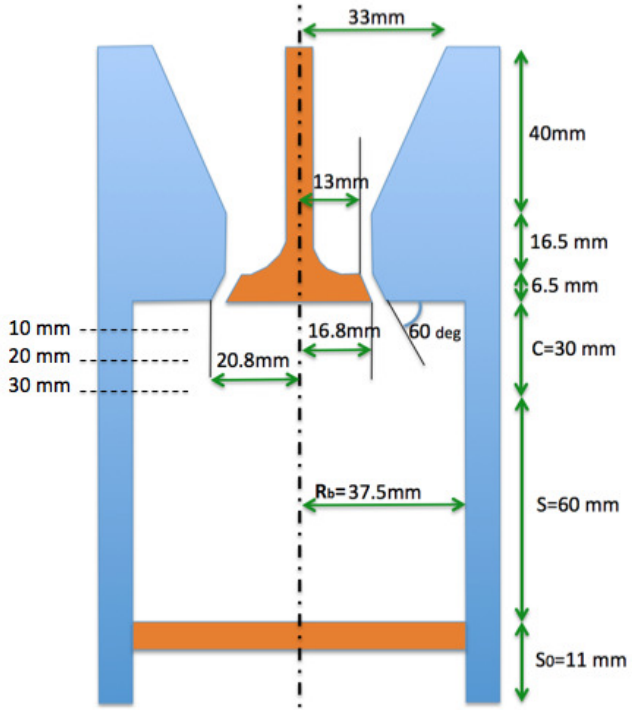

FIGURE 4.8: Engine geometry with three sampling lines at 10 mm, 20 mm, and 30 mm below the central valve (Morse et al. [155]) (Reprinted from Nguyen et al. [156] with permission of Springer).

depicted in Fig. 4.8. Additionally, the in-cylinder flow was driven by the moving piston that made the case more challenging. This test case examined the combination between the Lagrangian particles and the IBM for handling the stationary- and the moving boundaries in one simulation setup, in which: (1) a three-dimensional engine geometry with a fixed central valve was described by the IBM; (2) the modelling of the moving boundary using particle-based method was included to simulate the interaction between the fluid flow and the moving piston; (3) the inlet- and outlet flow conditions were calculated by the non-reflecting boundary conditions.

The motion of the piston was governed by a harmonic function:

$$z_p = s_1 + \frac{s}{2}\cos(\omega t) \quad , \tag{4.4}$$

where s represents the stroke length of the cylinder and the parameter $s_1 = s_0 + s/2$ (see Fig. 4.8). The Reynolds number in this test case is calculated using the engine speed ω, the stroke length s, the bore radius R_b and the molecular viscosity ν:

$$\mathrm{Re} = \frac{\omega s R_b}{\nu} \quad . \tag{4.5}$$

In the experiment by Morse et al. [155], the engine was motored at a speed of 200 RPM ($\omega = 21$ rad/s) where the mean piston speed ($\overline{V}_p = 0.4$ m/s) was considered to be small. Clearly, with the small Reynolds number of 2000, the test case would be time-consuming for the fully compressible solver. Therefore, in order to speed up the simulation and to still maintain the Reynolds similarity, both the engine speed ω and the molecular viscosity ν were increased by a factor of 10.

The simulation was carried out on a computational domain consisting of 7.9 million cells with the cell size of $\Delta = 0.5$ mm using 240 CPUs. Five consecutive engine cycles (360 crank angle degree (CAD) per cycle) were performed. Figure 4.9 shows the instantaneous vertical velocity in the tumble plane of four consecutive engine cycles.

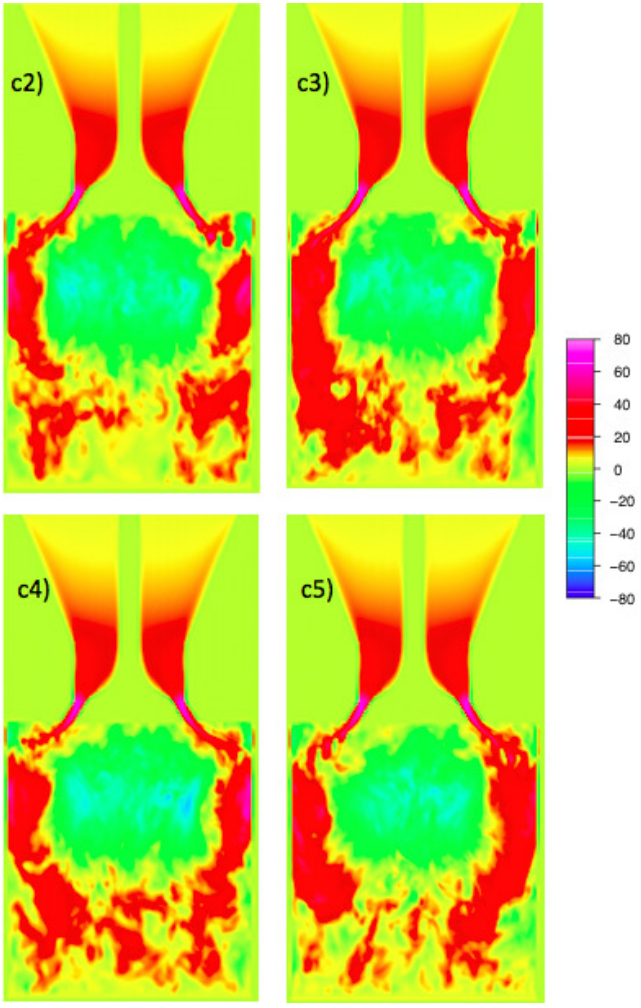

FIGURE 4.9: Instantaneous vertical velocity in the tumble plane of four consecutive engine cycles (c2 to c5) at 144 CAD.

The phase-averaged vertical velocity and RMS obtained from the simulation were compared against the experimental data at three sampling lines shown in Fig. 4.8 for 36° CA

and 144° CA after top dead center (TDC). As it is shown in Figs. 4.10 and 4.11, the velocity profiles and the corresponding RMS at three sampling lines (10 mm, 20 mm, 30 mm below the central valve) were overall well-captured by the simulations for both 36° and 144° CA. Only a small discrepancy between the predicted RMS and the experimental data was observed. Good agreement between the numerical results and the experimental data confirms the accuracy and the efficiency of the proposed method for the engine simulation. Further validations of this approach can be found in the work of Nguyen et al. [157, 156].

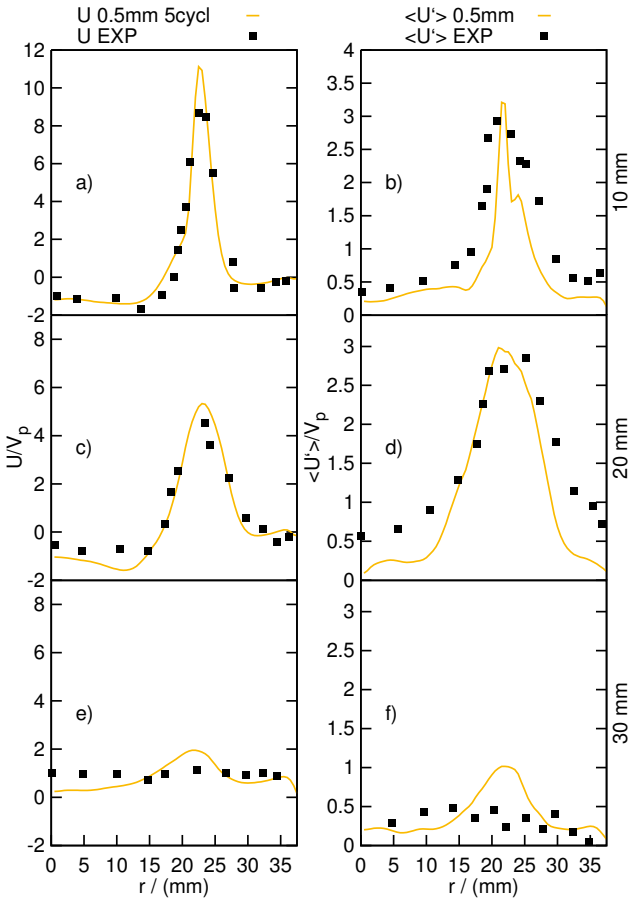

FIGURE 4.10: Predicted and measured velocity profiles at 36° CA: Vertical velocity at 10 mm (a), 20 mm (c) and 30 mm (e). The corresponding RMS at 10 mm (b), 20 mm (d) and 30 mm (f) (Reprinted from Nguyen et al. [156] with permission of Springer).

All the figures and the comparison between the obtained simulation results and the experimental data [155] are reprinted from the paper of Nguyen et al. [156] with permission of Springer.

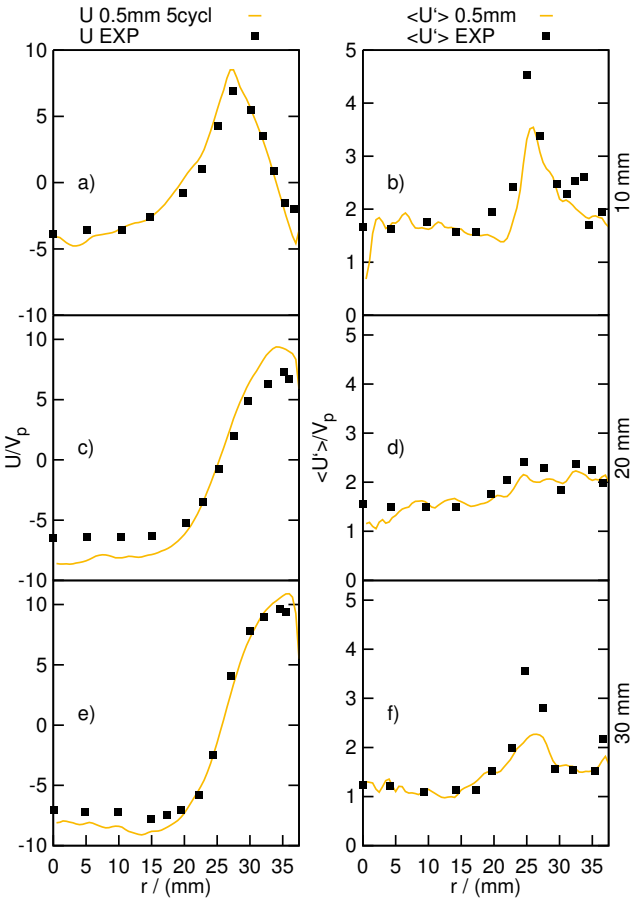

FIGURE 4.11: Predicted and measured velocity profiles at 144° CA: Vertical velocity at 10 mm (a), 20 mm (c) and 30 mm (e). The corresponding RMS at 10 mm (b), 20 mm (d) and 30 mm (f) (Reprinted from Nguyen et al. [156] with permission of Springer).

Chapter 5

LES of flow processes in an SI engine

Authors: T. Nguyen, P. Janas, T. Lucchini, G. D'Errico, S. Kaiser, A. Kempf
This chapter including all figures and tables was previously published in SAE International in United States, LES of Flow Processes in an SI Engine Using Two Approaches: OpenFoam and PsiPhi, SAE Technical Paper 2014-01-1121, 2014, T. Nguyen, P. Janas, T. Lucchini, G. D'Errico, S. Kaiser, A. Kempf. The author T. Nguyen and P. Janas developed the code, ran all the simulations, wrote the paper and generated all figures and tables. The authors T. Lucchini and G. D'Errico contributed to the development of the numerical methods in OpenFOAM. The author S. Kaiser provided the experimental data and the author A. Kempf contributed corrections, discussions and proof-reading.

5.1 Abstract

In this study two different simulation approaches to large eddy simulation of spark-ignition engines are compared. Additionally, some of the simulation results are compared to experimentally obtained in-cylinder velocity measurements. The first approach

applies unstructured grids with an automated meshing procedure, using OpenFoam and Lib-ICE with a mapping approach. The second approach applies the efficient in-house code PsiPhi on equidistant, Cartesian grids, representing walls by immersed boundaries, where the moving piston and valves are described as topologically connected groups of Lagrangian particles. In the experiments, two-dimensional two-component particle image velocimetry is applied in the central tumble plane of the cylinder of an optically accessible engine. Good agreement between numerical results and experiment are obtained by both approaches.

5.2 Introduction

Direct injection, downsizing and advanced combustion modes are key fuel-saving technologies in gasoline engines. To further decrease the fuel consumption and pollutant emissions and to increase the power output, a better understanding of the in-cylinder processes is crucial. Currently, advanced combustion modes cannot be used over the full operating range, often due to turbulence-induced flame quenching or as a result of poor fuel-air mixing near the spark. In-cylinder phenomena are commonly studied in single-cylinder research engines with optical access for laser diagnostics. On the other hand, engines are investigated by numerical techniques like CFD, which is often less expensive and more flexible than an experiment. As the state of the art, U-RANS simulations are successfully applied by industry to gain an understanding of the engine, but U-RANS will normally fail to predict cyclic variations. A promising alternative are large eddy simulations (LES) that resolve smaller flow structures, enabling them to capture cyclic variations. However, LES is computationally more expensive and requires high-quality meshes on which high-order numerical schemes must be applied. In the context of LES, several CFD codes like AVBP [45, 61], KIVA [158], FLUENT [159], or Star-CD [160] have shown at least partial ability to predict some relevant phenomena in internal combustion engines. A critical problem with the application of LES is that any discretization of less than second order accuracy and CFL numbers greater than one lead to artificial dissipation – causing slow mixing, insufficient flame wrinkling, and hence slow flame propagation. Unfortunately, it is very hard to satisfy these accuracy requirements with CFD codes that have been optimized for RANS on unstructured grids.

In this study, two different approaches are compared, which satisfy the stringent requirements for mesh quality and numerical accuracy for the LES of internal combustion engines. Both methods require very limited effort for the grid generation (less than one personnel hour for meshing). The first approach (OpenFOAM) [161] uses unstructured grids with deformable meshes. The second approach (PsiPhi) [162] is based on a structured grid with a combination of Lagrangian particles [163] and immersed boundaries [164] to represent the moving parts of the engine. Both codes, OpenFOAM and PsiPhi (in-house, developed at the chair of Fluid Dynamics, University Duisburg-Essen) were available without excessive cost for licensing and have demonstrated good parallel scaling, in the case of PsiPhi beyond 4000 cores. So far, the simulations have concentrated on a motored case, for which the velocity fields obtained in both approaches will be compared to each other and to measurements in an optically accessible engine. At the end of the paper, preliminary results are presented for a fired case.

5.3 Numerical methods

The filtered governing equations for mass (5.1), momentum (5.2), and total energy (5.3) are:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{u}_i\right) = 0 \tag{5.1}$$

$$\frac{\partial \overline{\rho}\widetilde{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{u}_i\widetilde{u}_j\right) = -\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial \widetilde{\sigma_{ij}}}{\partial x_j} - \frac{\partial}{\partial x_j}(\tau_{ij}^{sgs}) \tag{5.2}$$

$$\frac{\partial \overline{\rho}\widetilde{E}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\rho}(\widetilde{E}+\overline{p})\widetilde{u}_i\right) = \frac{\partial}{\partial x_j}\left(\kappa\frac{\partial \overline{T}}{\partial x_j}\right) + \frac{\partial}{\partial x_j}(\widetilde{\sigma_{ij}}\widetilde{u}_i) - \frac{\partial}{\partial x_j}(q_j^{sgs}) \tag{5.3}$$

where the gas density, the fluid velocity vector, the pressure, and the total energy are represented by $\overline{\rho}, \widetilde{u}, \overline{p}$ and $\widetilde{E}$, respectively. The temperature and the stress tensor are denoted as $\overline{T}$ and $\widetilde{\sigma}_{ij}$. The unresolved stresses τ_{ij}^{sgs} are computed by the standard Smagorinsky model [84]. The unresolved heat fluxes are represented by q_j^{sgs}.

The state of the gas is computed by the relations:

$$p = \rho RT, \quad c_v = R/(\gamma - 1), \quad c_p = c_v + R \tag{5.4}$$

The specific heat capacity at constant pressure and constant volume are indicated by c_p and c_v, respectively.

The total energy and the stress tensor of a Newtonian fluid can be expressed as:

$$\overline{\rho}\widetilde{E} = \frac{1}{2}\overline{\rho}\widetilde{u}_k\widetilde{u}_k + \frac{\overline{p}}{\gamma - 1} \tag{5.5}$$

$$\widetilde{\sigma}_{ij} = \widetilde{\tau}_{ij} - \overline{p}\delta_{ij} \tag{5.6}$$

$$\widetilde{\tau}_{ij} = \mu\left(\frac{\partial\widetilde{u}_i}{\partial x_j} + \frac{\partial\widetilde{u}_j}{\partial x_i}\right) - \frac{2}{3}\mu\frac{\partial\widetilde{u}_k}{\partial x_k} \tag{5.7}$$

The viscosity of the fluid is calculated from the reference temperature T_0, the viscosity μ_0 at reference temperature, and the Sutherland temperature T_s according to the Sutherland law:

$$\mu(T) = \mu_0\left(\frac{T}{T_0}\right)^{3/2}\frac{T_0 + T_s}{T + T_s} \tag{5.8}$$

5.4 Approach 1: OpenFOAM, unstructured grid

Codes for unstructured grids - like OpenFOAM - permit the efficient use of body conforming grids to represent complex geometries. In our case, the bottleneck in OpenFOAM is the mesh motion for large valve and piston displacements. The mesh inside the engine has to move according to the motion of the piston and valves without reducing the quality of the mesh. Here, we apply a mapping approach, in which the entire engine cycle (720 CA°) is split into intervals for which individual grids are used. Within each interval, the mesh maintains high quality while the grid points are moved. At the end of each interval, the results are mapped onto the next grid. This mapping approach has already been tested for U-RANS engine simulations [165] and was developed further

with the present meshing procedure. All grids are automatically generated at the start of the simulations and can be reused in later cycles.

5.4.1 Meshing with OpenFOAM

OpenFOAM's internal automated mesher, snappyHexMesh, is used to generate hexahedral cells. It reads a geometry input file of the engine (Stereo Lithography format - STL) and a block-structured mesh that surrounds the engine geometry, which is then wrapped to the STL geometry. Figure 5.1 shows the STL representation of the engine with the surrounded block mesh. The initial grid spacing of the block is 0.5 mm, which is also

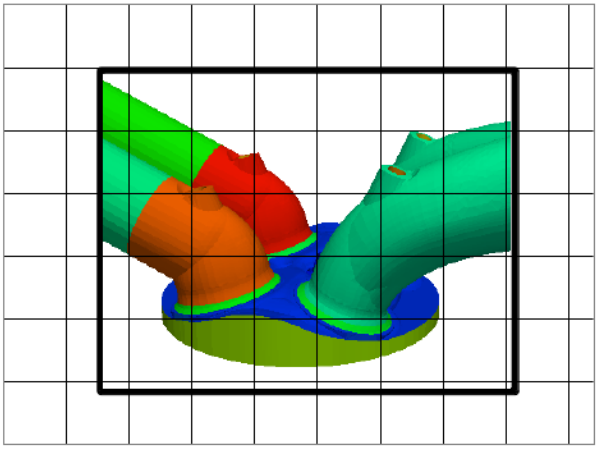

FIGURE 5.1: Sketch of the overlapped grid on the STL representation of the engine at top dead center (each color stands for a different patch) -OpenFOAM.

the biggest cell size in the engine. In order to capture details smaller than 0.5 mm, it is necessary to divide the STL file of the engine in different patches, which can be assigned within snappyHexMesh and used for local refinement. SnappyHexMesh refines the cells in the vicinity of the patch by repeatedly splitting a hexahedral cell into eight cells. The smallest discretized valve gap was set to 0.5 mm; for this position a refinement has been applied twice, leading to a cell size in the valve seat gap of 0.125 mm as shown in Fig. 5.2. In total, 39 different patches have been created for the STL representation of the engine. An automated tool for generating STL files for a full cycle has been developed based on Lib-ICE [166, 167, 168, 169] technologies. It reads a STL file of the engine at top dead center (TDC) together with the minimum valve lift position, the valve lift profiles, the stroke, and the connecting-rod length of the engine. To enforce complete valve closure, either internal walls are added as curtains around the valve seat or the ports

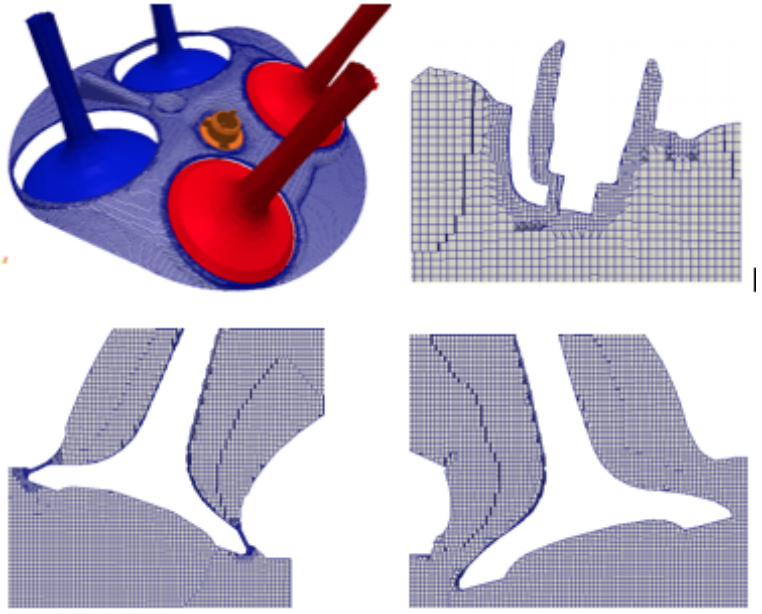

FIGURE 5.2: Top left: surface grid of the cylinder head with intake (blue) and exhaust (red) valves. Top right: mesh around the spark plug (cell size 0.25 mm) Bottom: cut through the exhaust valve in closed position (cell size inside the gap 0.125 mm, left) and cut through the intake valve at a lift of 5.5 mm - OpenFOAM.

are completely detached from the cylinder volume. The former method is computationally less expensive, but cannot resolve the dynamics within the port. For a multi-cycle simulation, the valves should be closed with internal walls, in order to keep track of the flow dynamics in the ports. A total of approximately 100 different grids have been used to represent the intake, compression and expansion stroke. The cell size varied from 5.3 Mio to 0.5 Mio (see figure 5.4). This meshing approach with OpenFOAM is generic

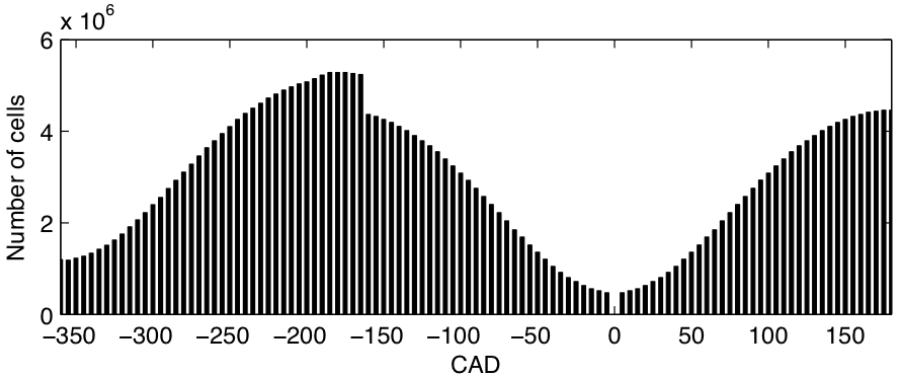

FIGURE 5.3: Number of cells used for the intake, compression and expansion stroke (intake port detached at -160 CA°) -OpenFOAM.

and can be applied to any combustion engine. Furthermore, parts of the engine can be added easily (e.g. the spark plug) or removed (ports, for valve closure). The entire

workflow is automated, except for the creation of the basic STL file of the engine. It takes a maximum of one hour on 48 cores to generate the grid for a given interval.

5.4.2 Mesh motion with OpenFOAM

For the motion of the internal grid points, a Laplace equation is solved for the cell-velocity field [170]. The mesh-motion Laplace equation reads:

$$\frac{\partial}{\partial x_i}\left(\Gamma \frac{\partial u_{cell,i}}{\partial x_i}\right) = 0 \tag{5.9}$$

where Γ is a diffusion constant for the mesh motion field and u_{cell} is the velocity vector of the cell itself. A time varying boundary condition based on the velocity of the piston and valves is applied for the moving parts. The resolved cell-velocity field is then used to determine the new position of the cell according to the following equation:

$$x_{new} = x_{old} + u_{cell}\Delta t \tag{5.10}$$

The mesh diffusivity can be constant or dynamic. Here, the diffusion field is decreased near moving boundaries, which creates artificial stiffness, so that cells are less deformed near the moving boundaries.

5.4.3 Numerics with OpenFOAM

The numerical solver is pressure-based, such that it can handle the coupling of implicitly discretized time-dependent flow equations. For the pressure-velocity coupling the PISO-scheme [171] is applied with the following semi-discretized pressure equation [172]:

$$\frac{\partial \Psi p}{\partial t} + \frac{\partial \Psi p u_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\frac{\rho}{\mathcal{A}}\left(\frac{\partial p}{\partial x_j}\right)\right] = 0 \tag{5.11}$$

with

$$\Psi = \frac{\rho}{p}, \quad u = diag(\mathcal{A})(\mathcal{A})^{-1} \tag{5.12}$$

where $\mathcal{A}$ represents the coefficient matrix of the linearized algebraic function of the momentum equation (5.2). The PISO approach permits to use larger time steps once

the valves are closed, so that the computational cost is similar to the fully compressible method used in PsiPhi.

The transport equation for the enthalpy reads:

$$\frac{\partial(\rho h)}{\partial t} + \frac{\partial(\rho h)u_i}{\partial x_j} - \frac{\partial}{\partial x_j}\left[\alpha_{eff}\left(\frac{\partial h}{\partial x_j}\right)\right] = \frac{\partial p}{\partial t} + u_i\frac{\partial p}{\partial x_j} \tag{5.13}$$

with α_{eff} as the effective thermal diffusivity accounting for the local mean and turbulent diffusivities.

The flow field is advanced in time by an implicit, second-order backward scheme. The convective scalar fluxes are discretized by a TVD scheme, using the Sweby limiter [125]. For the momentum equation, a switch between the CDS and the TVD (Sweby) schemes based on the local Mach number is used: for small valve lifts, the flow in the valve gap can reach high velocities (Ma=1) introducing numerical instabilities that must be avoided by applying a TVD scheme. At the same time, the TVD scheme would increase numerical dissipation over CDS by a degree that is not acceptable with LES. We achieve a good balance between accuracy and low dissipation by switching from CDS to TVD wherever the Mach number exceeds 0.5. The effect is illustrated by fig. 4, showing the maximum Mach number over the crank angle for the intake stroke: in the interval between -390 CA° and -385 CA°, very high velocities occur in the small valve gap of 0.5 mm. It can be seen that the CDS scheme causes strong spikes, whereas the CDS-TVD scheme gives a smoother time course of the velocity inside the valve gap.

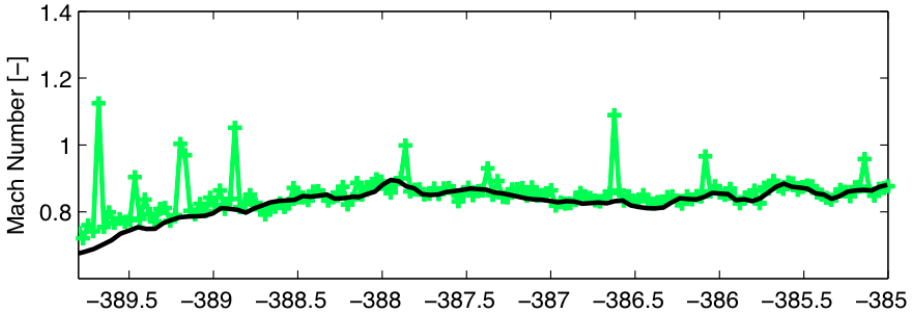

FIGURE 5.4: Plot of the maximum Mach number during the intake stroke (-390 CA° to -385 CA°); (-+-): CDS for the momentum equation, (–): CDS-TVD scheme for the momentum equation - OpenFOAM.

5.5 Approach 2: PsiPhi, immersed boundaries

In this approach, the immersed-boundary technique is employed for the representation of the complex geometry of the engine. First, volumetric pixels (voxels) of different 3D engine components are generated from the corresponding 3D CAD models of the cylinder, the intake port, the outlet port and the cylinder head. These voxels are then mapped to the main Cartesian equidistant grid to build the immersed boundaries of the simulated engine. This approach yields high numerical accuracy, permits more cells, promises high model accuracy (isotropic cells), and ensures that finer grids are maintained in the region of the flame, while being cheaper at the same CFL number. The method is available in the in-house code PsiPhi, which has been developed to combine high computational accuracy, good parallel scaling and high efficiency on state of the art computer hardware. Focusing on the LES of combustion, PsiPhi has proven its abilities in simulations of fully premixed, partially premixed and non-premixed flames [103, 162, 173, 174] of gas jets [175], and pulverized coal [176].

5.5.1 Mesh generation with PsiPhi

Mesh generation in PsiPhi is simple, flexible, and fast. It takes less than an hour to generate any computational grid with high resolution from the CAD model, for ten to hundreds of millions of cells. The generated voxels from the moving parts including the piston, the intake and exhaust valves are used to create the different groups of particles that describe the moving objects. Figure 5 shows engine valves meshed with this method. Note that the sharp voxel-edges and voxel-corners displayed in fig. 5.5 are not "seen" by the simulation. The stairs are only a result of the graphical representation of the cells; they do not induce any flow over a stair-like surface (the stairs can be considered as a first order approximation of the flat boundary).

The following section is dedicated to the moving-boundary technique that is applied within this study. With this approach, the simulated engine can be decomposed into structured cubic domains that are highly effective for parallelization.

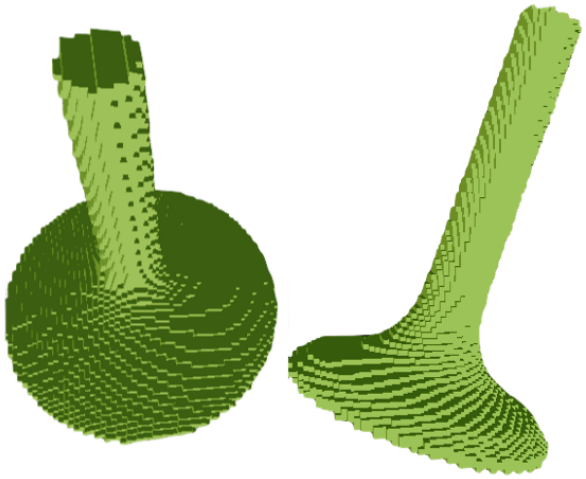

FIGURE 5.5: The intake and exhaust valves are represented by Lagrangian particles and mapped to voxels on the computational grid (0.5 mm cell size) - PsiPhi.

5.5.2 Moving boundaries with PsiPhi

Coupled to the immersed boundary method, a mesh-free approach is employed to handle the moving boundaries of the engine simulation. Moving parts like the piston and the valves are formed by different groups of Lagrangian particles. These particles are controlled by the same law of motion as the corresponding moving objects. Using this method, no mesh alteration is required, which avoids the difficulty involved in the generation of high quality moving meshes.

In this mesh-free technique, the momentum equation is modified to account for the particle force (F_p) that is used to impose the flow condition at the boundary of any moving objects. The resulting momentum equation reads:

$$\frac{\partial \overline{\rho}\widetilde{u_i}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{u_i}\widetilde{u_j}\right) = -\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial \widetilde{\sigma_{ij}}}{\partial x_j} - \frac{\partial}{\partial x_j}(\tau_{ij}^{sgs}) + F_p \tag{5.14}$$

Calculating the particle force F_p requires the imposed velocities and the imposed pressure at the boundary between the solid phase and the gas phase. These velocities and pressure are computed from the particle velocity Vp_i and the velocity of neighbouring fluid cells Vf_i [163]:

$$Vs_i = 2Vp_i - Vf_i \tag{5.15}$$

In the case of more than one neighbouring fluid cell, the velocity of a solid cell is computed from the particle velocity Vp_i and the average of the neighbouring fluid cells:

$$Vs_i = 2Vp_i - \frac{\sum_j^{Nb} Vf_{ij}}{Nb} \tag{5.16}$$

The pressure of the solid cells at the boundary can be computed based on the pressure gradient that is calculated from the density ρ of the neighbouring fluid cell and the acceleration of the particle a_{pi} [163]:

$$\frac{\partial p}{\partial x_i} = -\rho a_{pi} \tag{5.17}$$

With more neighbouring fluid cells, the pressure gradient of the solid cell can be evaluated:

$$\frac{\partial p}{\partial x_i} = -\frac{\sum_j^{Nb} \rho_j}{Nb} a_{pi} \tag{5.18}$$

Where ρ_j is the density of the neighbouring fluid cell j and Nb denotes the number of the neighbouring fluid cells.

In this approach, the activation and deactivation of the mesh cells are handled by the flux-blended scheme, which is a combination between the upwind and the central differencing scheme. When the cell changes from solid to fluid, a linear interpolation scheme with weighting factors is used to create the history in the fresh cleared cells from the momentum of the neighbouring fluid cells.

5.5.3 Non-reflecting boundary conditions

In compressible flow, the truncated inlet or outlet usually leads to numerical oscillations resulting from the spurious waves, since the physical boundary conditions are not known in detail. In engine simulations, correctly imposing the physical boundary conditions would require to include the long intake and exhaust ports, which would increase the computational effort by a considerable amount. In this study, the inlet and outlet ports are kept short, and Navier-Stokes characteristic boundary conditions (NSCBC) were implemented to compute the numerical conditions on the inlet and outlet planes. The amplitude variations of the incoming and outgoing waves were determined from the

"Local One-Dimensional Inviscid (LODI) relations" for the primitive variables at the boundary [136].

5.6 Engine Geometry

The single cylinder gasoline direct injection engine of the Institute for Combustion and Gas Dynamics (University of Duisburg-Essen) features four valves and a pent-roof head. It provides optical access through the piston crown and the cylinder liner. Table 5.1 shows the key specifications of the engine.

Disipalcement	499 cm^3
Stroke	90 mm
Bore	84 mm
Connecting rod length	161 mm
Compression ratio	10:1
Number of valves	4
Exhaust valve open	180 CA°
Exhaust valve closed	400 CA°
Intake valve open	-405 CA°
Intake valve closed	-145 CA°

TABLE 5.1: Specification of the engine. Top dead center (TDC) of the compression stroke is taken to be zero degrees crank-angle (0 CA°).

Pressure is measured by crank-angle-resolving sensors. Piezoresistive sensors determine the pressure in the intake and exhaust ports 100 mm upstream of the valves, and a piezoelectric one the relative in-cylinder pressure near the spark-plug location. The in-cylinder pressure was pegged to the absolute intake pressure at a position near bottom dead center (BDC).

5.7 Measurements

The engine was motored at a speed of 1000 rpm; intake, coolant and oil temperatures were kept at 333 K. The maximum intake valve lift was 7 mm at -270 CA°, while the maximum exhaust valve lift was 3.5 mm at 280 CA°. The instantaneous velocity field in the tumble symmetry plane was measured by particle image velocimetry (PIV). Figure 5.6 illustrates the optical access through a quartz cylinder and a quartz window in the

piston crown. This results in a view of about 60 mm width, extending about 30 mm down from the fire deck.

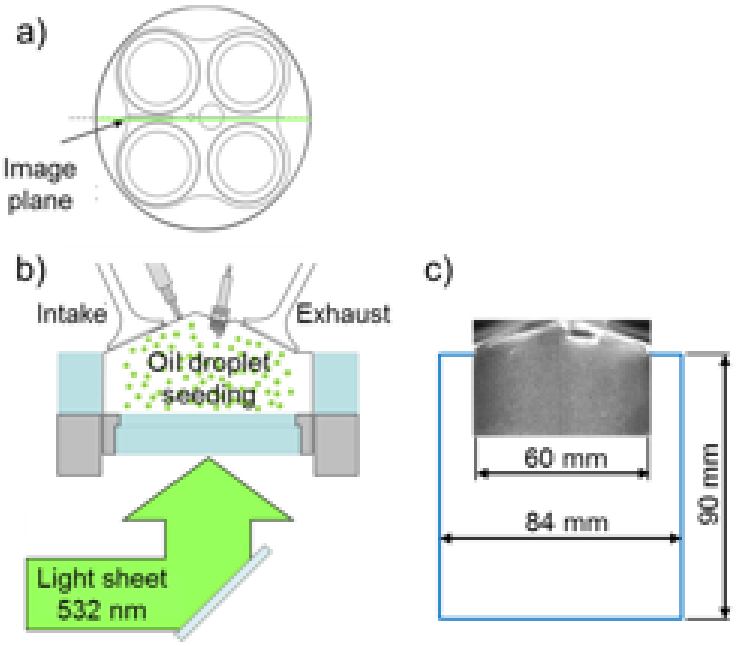

FIGURE 5.6: (a) bottom view of the engine head, (b) side view of the head, cylinder and measurement domain, (c) field of view and dimensions of the cylinder.

The intake-air was mixed with silicon-oil droplets (nominal diameter 0.2 µm – 0.3 µm) in a chamber 500 mm upstream of the intake valves. The particles were illuminated with a double-pulsed Nd:YAG laser at 532 nm. The laser light sheet with a thickness of 0.9 mm was directed into the engine via a mirror in the bottom of the slotted, extended piston. The light scattered from the particles was detected by a double-frame CCD camera; each image pair was processed by LaVision Davis 7.2 to provide a two-dimensional velocity vector map. Spurious vectors were removed using standard validity checks [177]. At selected crank angles (-180, -140, -120, -90) 100 shots each were taken of the velocity field to be used for the validation of the two simulation approaches.

5.8 Simulation Setup

In both approaches the crevice volume between the liner and the piston is neglected, as well as a chamfer between the glass liner and cylinder head. To compensate for these crevice volumes, the clearance height at TDC was increased by 2.6 mm. In OpenFOAM with its body-fitted mesh, the minimum valve gap was chosen to 0.5 mm with a grid

spacing of 0.125 mm within this gap. In the second approach, PsiPhi, the valve gap was resolved by the same 0.5 mm cells that were used throughout the domain. Table 5.2 and fig. 5.7 show the computational domains and key parameters for both approaches.

Approach	**Mesh type**	**Grid spacing**	**Number of cells**
Unstructured (OpenFOAM)	Body-fitted	0.125-0.5mm	BDC: 5.3 Mio (with intake) TDC: 1.2 Mio (with intake)
Structured (PsiPhi)	Block-structured	0.5mm	BDC: 4.8 mio (Fluid) TDC: 0.48 Mio (Fluid) Total: 9.6 Mio (Fluid)

TABLE 5.2: Grid specification of the two approaches.

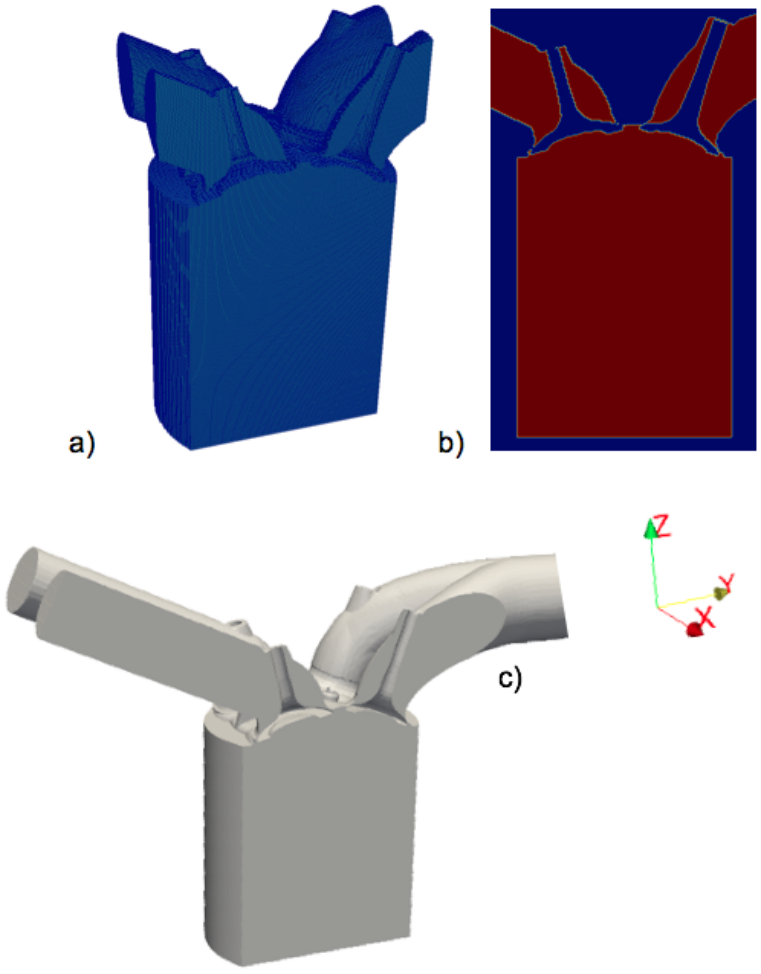

FIGURE 5.7: (a) Fluid cells at BDC for the immersed boundary approach, (b) Cut through the domain of the immersed boundary approach at BDC, red: fluid cells, blue: solid cells, (c) Computational domain of the unstructured approach (body fitted grid), left intake, right exhaust port.

5.9 Boundary conditions

The simulations were started at -360 CA°. At this position, the intake valves are already opened with a lift of 1.3 mm and the exhaust valves are about to close (0.26 mm). Because both approaches have a physical minimum valve gap of around 0.5 mm, the valve overlap was not taken into account here. In both approaches, the walls are treated as adiabatic with a no-slip condition applied. The intake valve closure was set to -160 CA°.

The velocity field for the first approach (OpenFOAM) was initialized as stagnant, for the second approach (PsiPhi) with artificial fluctuations smaller than 6 m/s. All other scalar variables were initialized uniformly within the entire domain. The temperature was set to 333 K and a pressure of 1.0165 bar was imposed, which was taken from the absolute intake pressure measurement. With the first approach (OpenFOAM), the measured pressure profile was imposed on the intake. With the second approach, the NSCBC [136] were applied, in which the inlet velocities are computed from the estimated time dependent mass flow and the imposed intake pressure is solved directly from the amplitudes of the characteristic waves coming into the computational domain.

5.10 Results

Simulations were carried out to compare the capabilities of the two approaches. The results are compared to the velocity measurements. Table 5.3 shows the computational effort required for the intake, compression, and expansion stroke.

Approach	**OpenFOAM**	**PsiPhi**
Number of CPUs	96	192
CFL number	1.0	0.3
Δt min [s]	0.7E-6	0.34E-6
Δt max [s]	0.275E-5	0.56E-6
Calculation time	9 days	10 days

TABLE 5.3: Simulation effort for intake, compression and expansion stroke.

As a result of the meshing scheme, in the PsiPhi approach the total number of cells is fixed during the entire simulation, whereas it is varying with the OpenFOAM approach, such that the number of CPUs also can be varied. In figure 5.8, the in-cylinder pressure

is plotted over the intake stroke. The calculated pressure is sampled near the spark plug location and compared to the measurements. The simulations were initialized with the measured intake pressure, which leads to a deviation of 50 mbar at -360 CA° relative to the measured in-cylinder pressure. The simulated pressure curves are matching each other after 45 CA°. Beyond this point, the simulated pressure curves divert and coincide again at -180 CA° (BDC). This deviation is expected from the different strategies for setting the boundary conditions at the intake port since OpenFOAM uses a time-varying pressure and PsiPhi employs non-reflecting boundary conditions.

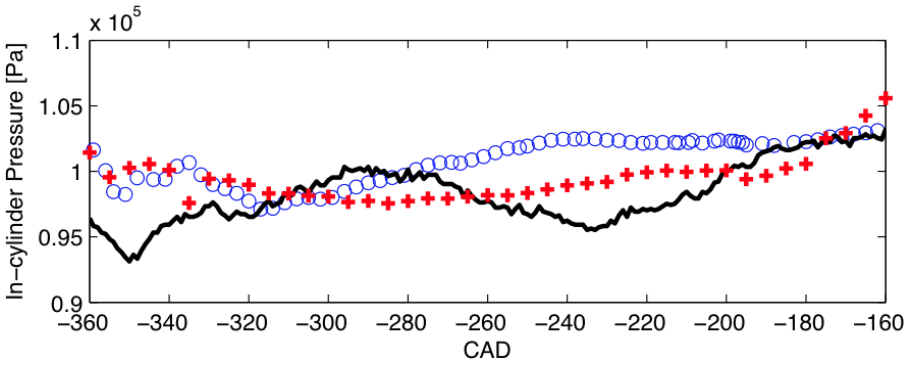

FIGURE 5.8: In-cylinder pressure during the intake stroke until IVC (-160 CA°) (Measurement (—), OpenFOAM (∘), PsiPhi (+)).

At the time of the intake valve closure (-160 CA°), the in-cylinder pressure from both simulations match the measured in-cylinder pressure. From this crank angle on, the piston compresses the air inside the cylinder. Figure 5.9 shows the in-cylinder pressures during the compression stroke. The small deviation of the pressure for the compression and expansion stroke may be related to the specific heat capacities and to the adiabatic boundary conditions. PsiPhi assumes constant heat capacity, while in OpenFOAM the heat capacity is a function of the temperature. The peak pressure computed with the first approach (OpenFOAM) is 25.4 bar, with the second approach (PsiPhi) it is 24.9 bar.

The in-cylinder gas temperature was computed from the measured pressure trace by the polytropic relation of an adiabatic system for air and compared to the temperature obtained from the simulations. Figure 5.10 shows good agreement between the simulated in-cylinder temperatures and the calculated polytropic temperature. However, matching a motored pressure trace is certainly not a very stringent test of the validity of the simulations, hence more detailed results will be examined in the following.

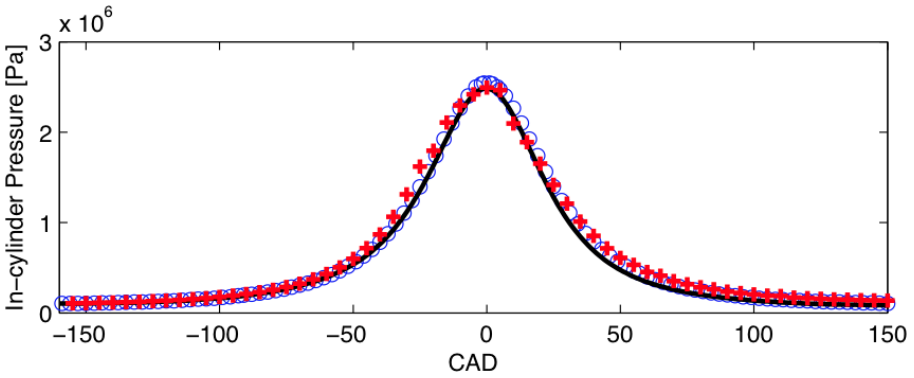

FIGURE 5.9: In-cylinder pressure during compression and expansion (measurement (—), OpenFOAM (∘), PsiPhi (+)).

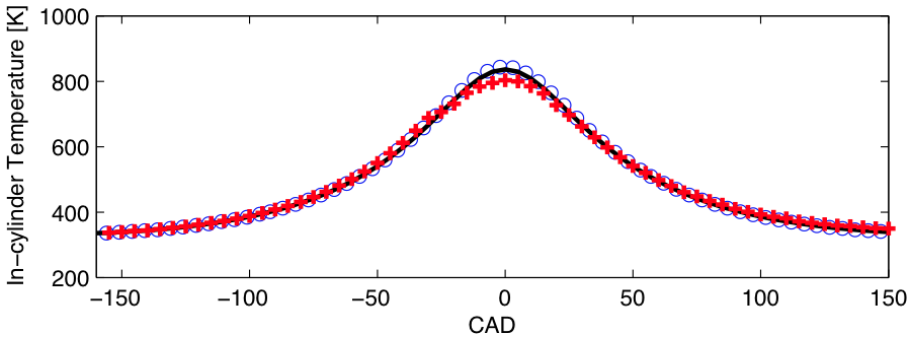

FIGURE 5.10: In-cylinder temperature during compression and expansion (Polytropic relation for temperature (—), OpenFOAM (∘), PsiPhi (+)).

The velocity fields for the z-component at -310 CA° (50 CA° after the start of the simulation) are shown for both approaches in fig. 5.11. The velocity profiles are sampled 20 mm below the cylinder head. Similar flow patterns in the two simulations throughout the intake stroke can be related to the high velocity flow through the valve gap, where most of the turbulence is generated, dominating the surrounding flow. However, it can be seen in fig. 5.11 (right column), that more small structures remain in the velocity component plots of the PsiPhi simulation, whereas the flow fields of the OpenFOAM calculation are smoother (left column fig. 5.11). This discrepancy is a result of the different numerical schemes: with OpenFOAM, a combination between the central difference scheme (CDS) and the total variation diminishing scheme (TVD) is applied for the momentum fluxes, inside the valve gap for Ma>0.5. By using the TVD scheme, kinetic energy partially dissipates, resulting in smoother flow fields. With PsiPhi, the momentum fluxes are discretised by the CDS scheme, resulting in less dissipation.

Figure 5.12 shows the tumble velocity profiles of the z components at 10 mm and 30 mm underneath the cylinder head for both simulations and the experiment. The tumble flow

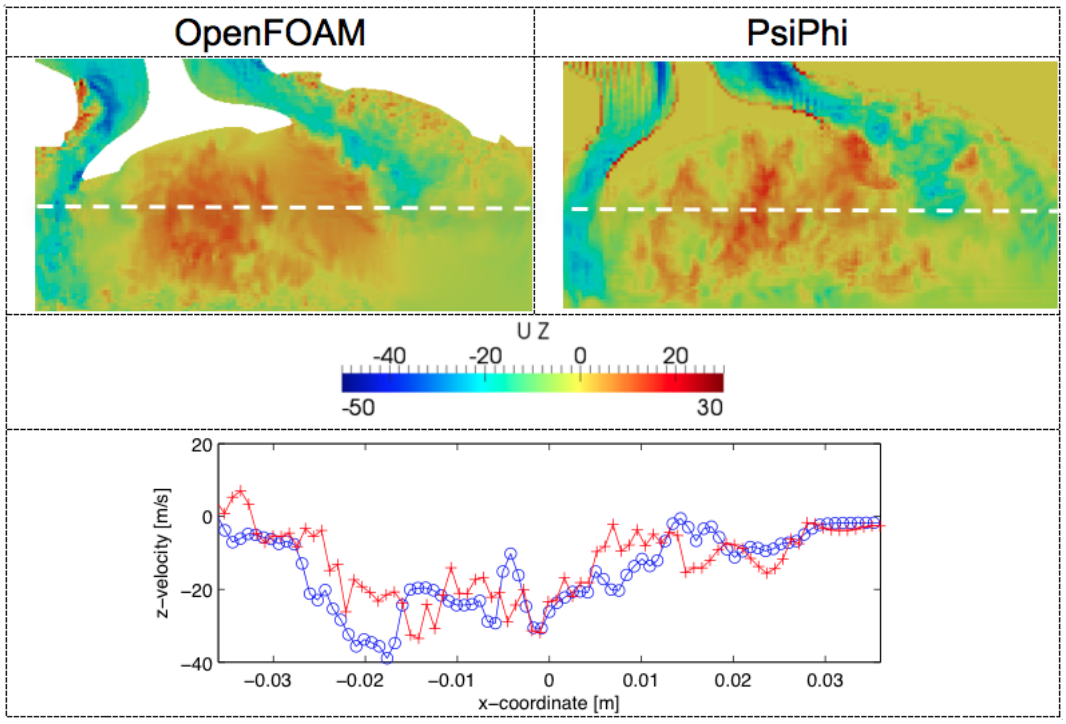

FIGURE 5.11: z-velocity component from both simulations at -310 CA°, at the position of the dashed white line, the velocity profile are taken (OpenFOAM (—○—), PsiPhi (—+—)).

velocities are qualitatively captured by both simulations. The instantaneous velocities in the lower cross-section (30 mm) at -180 CA° (fig. 5.12) show similar flow patterns for both approaches.

In fig. 5.13, the flow field during the late compression at -90 CA° is shown. The fine structures are preserved in both simulations, which imply the importance of low dissipative numerical schemes.

The two-dimensional cross-sections through the cylinder volume in fig. 5.12 do not reveal the full extent of the inhomogeneity of the velocity field at that location. Figure 5.14 shows horizontal cuts for both approaches in a plane 30 mm below the cylinder head at -180 CA°. Both approaches capture a large range of turbulent structures, with the overall field being similar.

Based on this work, further investigations will be carried out for the combustion phase in SI engines. Figures 5.16 and 5.15 present preliminary results of the premixed turbulent combustion performed with OpenFoam and PsiPhi, respectively.

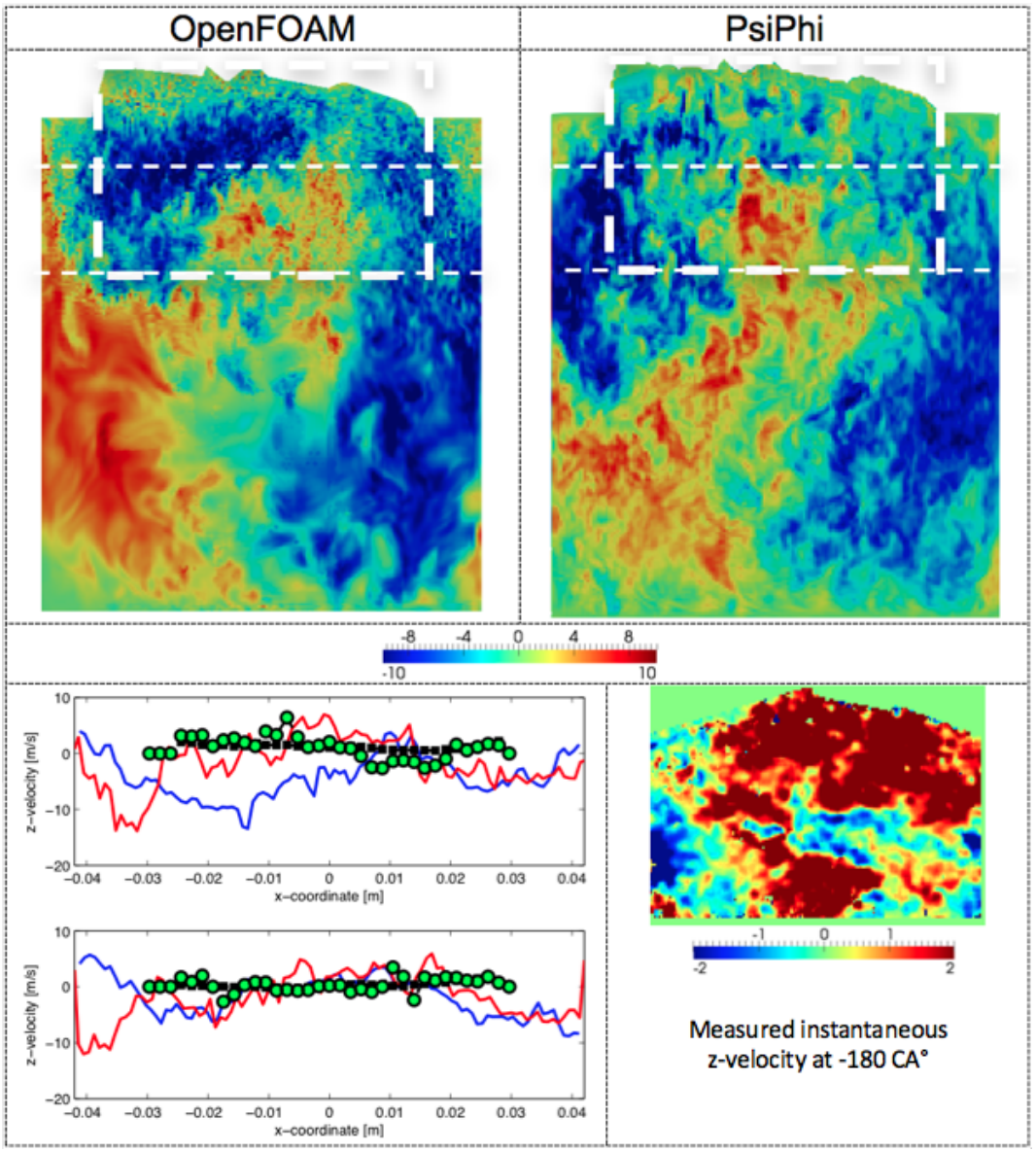

FIGURE 5.12: Comparison of the z-velocity components (m/s) of both simulations at -180 CA°: The white box denotes the window of the measurement at -180 CA°. The velocity profiles are sampled at 10 mm and 30 mm underneath the cylinder head (OpenFOAM (—), PsiPhi (—), an instantaneous measurement (•) and phase-locked average over 100 cycles (■).

5.11 Conclusion

In this paper, Large Eddy Simulation for an IC engine is studied with two different simulation tools. The simulation results show good qualitative agreement with each other and with experimental data.

Despite the differences in the meshing strategy, numerical schemes, boundary conditions and moving-mesh description, similarity of the numerical results can be observed and

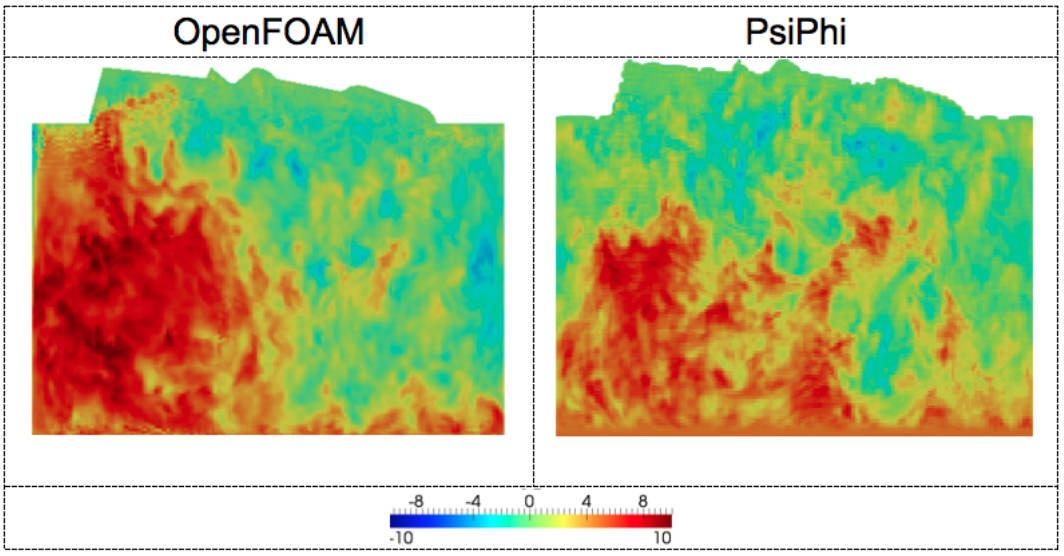

FIGURE 5.13: Comparison of the z-velocity components (m/s) from both simulations at -90 CA°.

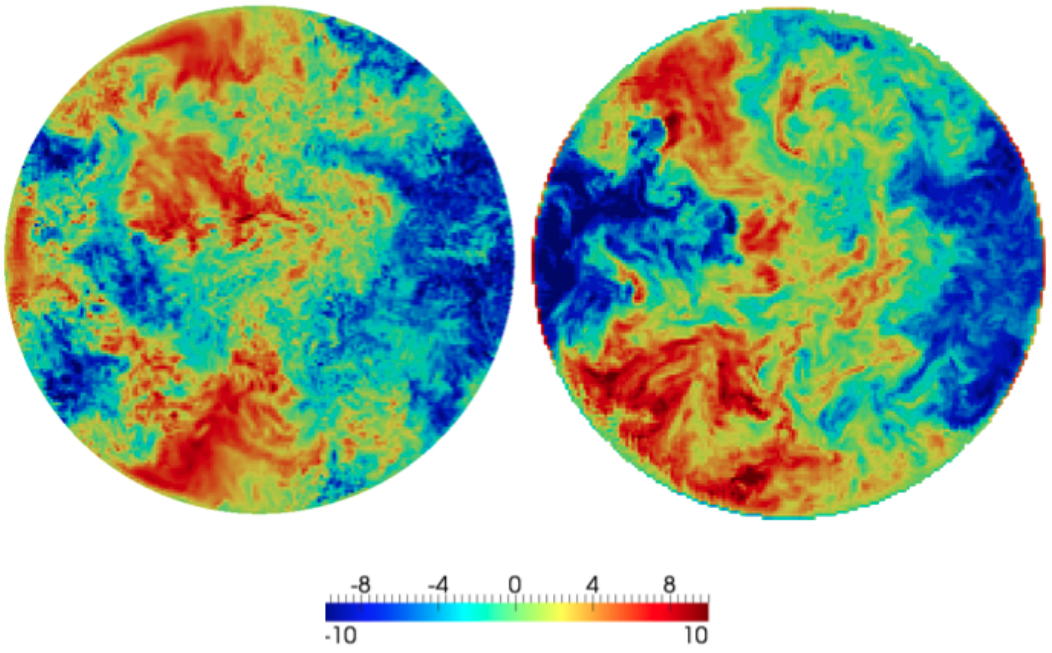

FIGURE 5.14: z-velocity in horizontal cross-section (m/s), 30 mm below the fire deck at -180 CA° (left: OpenFOAM, right: PsiPhi).

qualitatively compared. Similar flow patterns are obtained in the tumble flow and in the valve section at different crank angles for both simulations.

With the OpenFOAM approach, the mesh is moving within the domain, causing interpolation errors inside the domain - a problem that is avoided with PsiPhi. On the other hand, OpenFOAM can achieve better resolution within the valve gap due to the local mesh refinement. Using a central differencing scheme for the momentum transport with

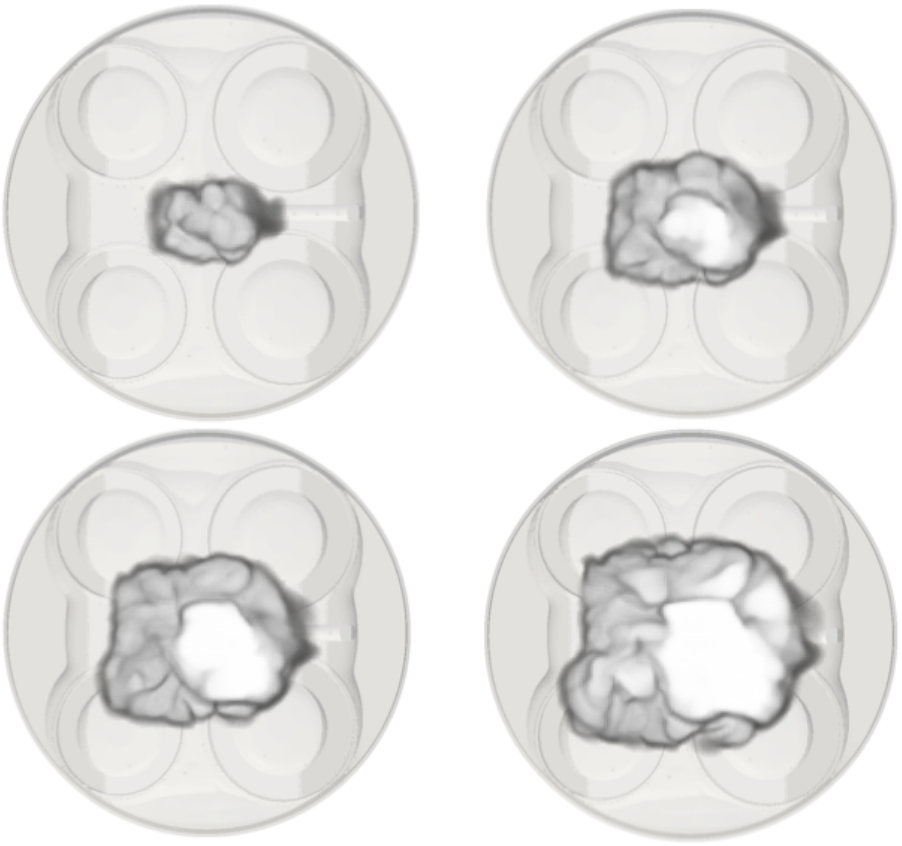

FIGURE 5.15: Volume rendering of the flame front propagation in the IVG engine (i.e., the engine that was used for comparing the cold flow in figs. 5.8-5.14) as seen from the top - OpenFOAM.

Mach-number dependent blending to TVD, both approaches achieve little numerical oscillation and manage to resolve relatively small turbulent structures throughout the computational domain.

Whereas PsiPhi is fast because of its simplicity and optimisation for modern hardware, OpenFOAM achieves a similar calculation time as the fully compressible scheme is replaced by the PISO scheme and a much bigger CFL number is used. The meshing of the geometry is automated in both approaches, such that the preparation of the setup files takes less than one personnel hour for each approach, starting out from an existing STL file. With PsiPhi, grid-generation completes within minutes, whereas with OpenFOAM, it takes around 2000 CPU hours for the geometry presented here, contributing roughly 10% to the computational cost for a single cycle. However, also with OpenFOAM the simulation can be started within one hour, i.e., as soon as the first grid has been generated.

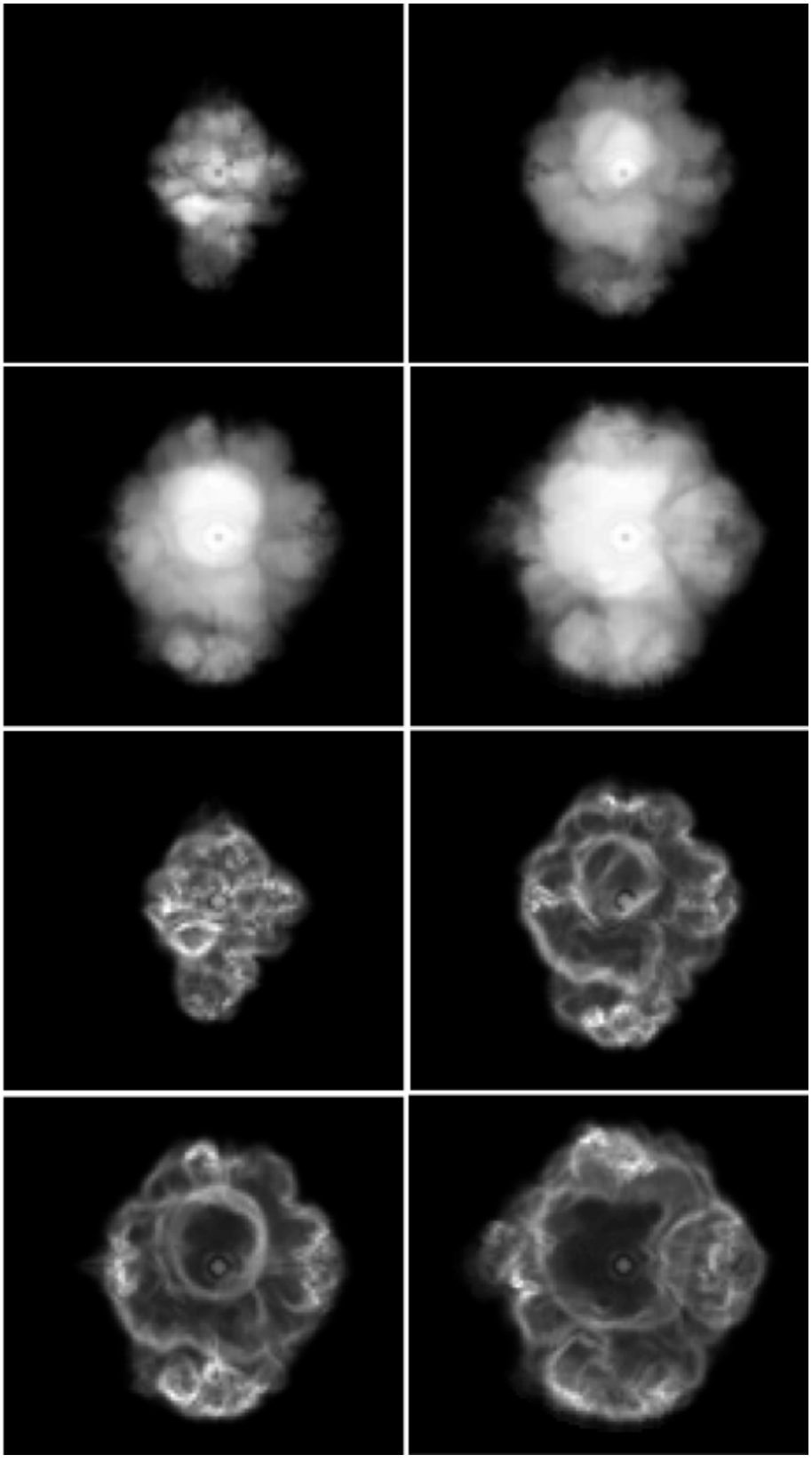

FIGURE 5.16: Volume rendering of the simulated flame propagation visualised by the progress variable (top four images) and the reaction source term (bottom four images) as seen from the top at different CA° for the optical engine [158] at the Center of Smart Interfaces (CSI), Technische Universität Darmstadt - PsiPhi.

5.12 Appendix

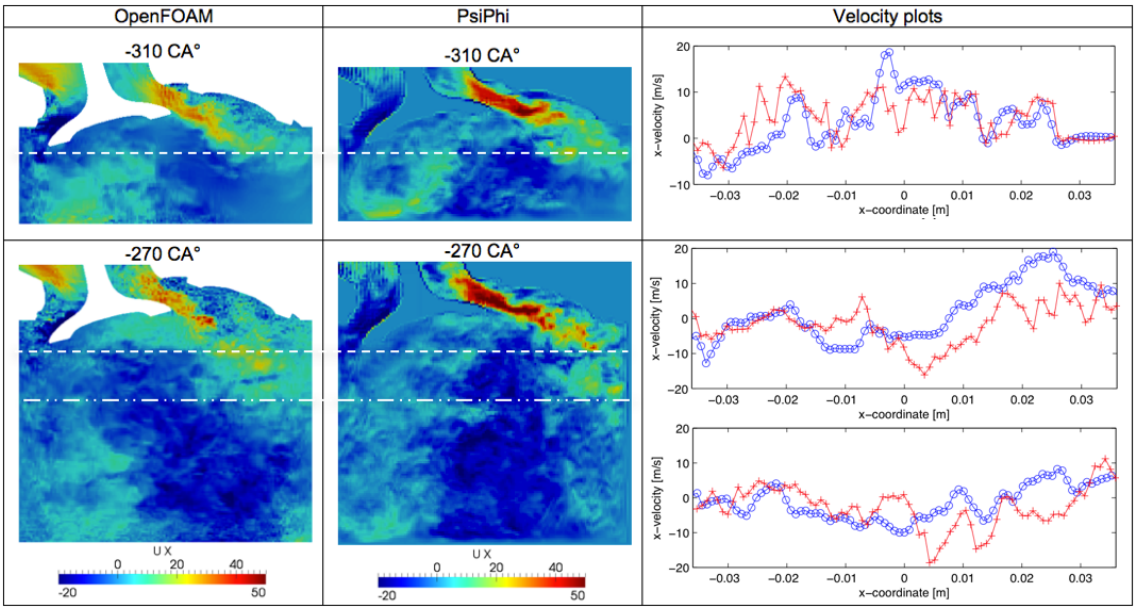

FIGURE 5.17: Contour plots of the x-velocity component in a cut through the intake valve at -310 CA° and -270 CA° and the corresponding velocity profiles for 10 mm(- -) and 30 mm (-···-) below the cylinder head.(OpenFOAM (—○—), PsiPhi (—+—)).

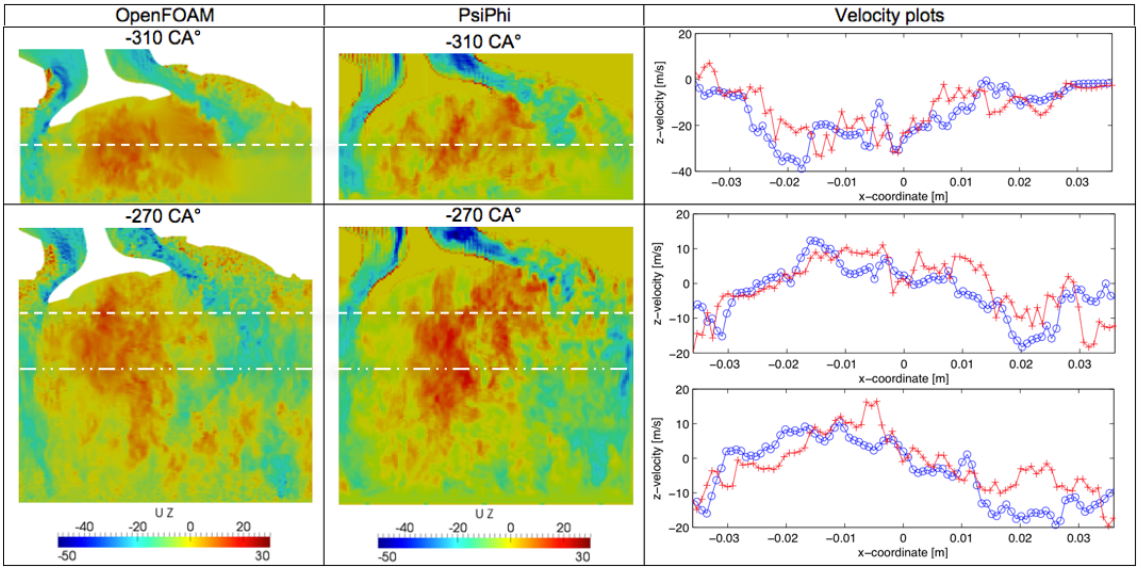

FIGURE 5.18: Contour plots of the z-velocity component in a cut through the intake valve at -310 CA° and -270 CA° and the corresponding velocity profiles for 10 mm(- -) and 30 mm (-···-) below the cylinder head.(OpenFOAM (—○—), PsiPhi (—+—)).

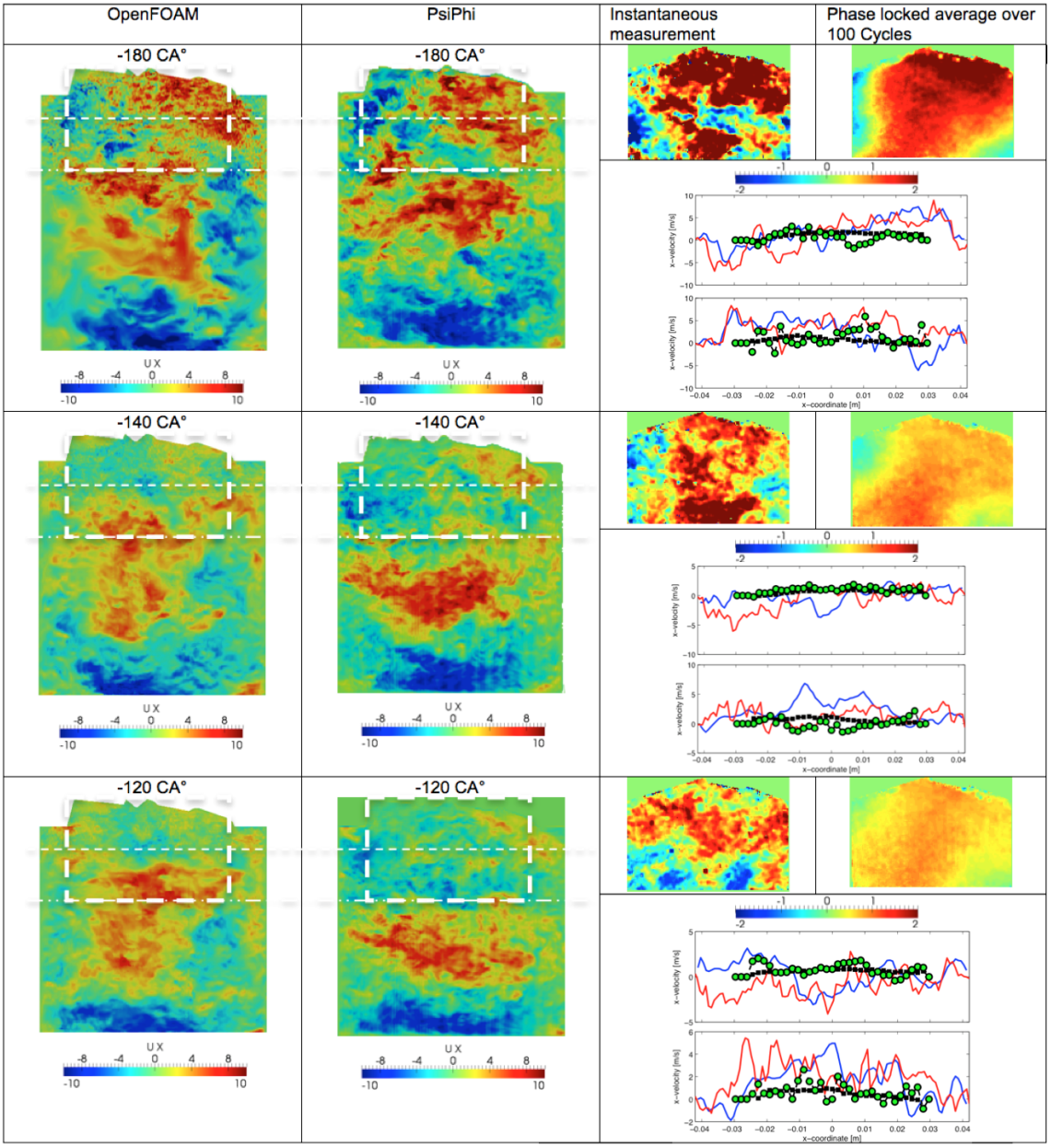

FIGURE 5.19: Contour plots of the x-velocity component of the tumble symmetry plane at -180 CA°, -140 CA° & -120 CA°. Velocity profiles for 10 mm(- -) and 30 mm (-···-) below the cylinder head. One measured instantaneous velocity contour plot is shown and the phase locked average from 100 cycles. (OpenFOAM (—○—), PsiPhi (—+—), instantaneous measurement (●) and phase-locked average over 100 cycles (■)).

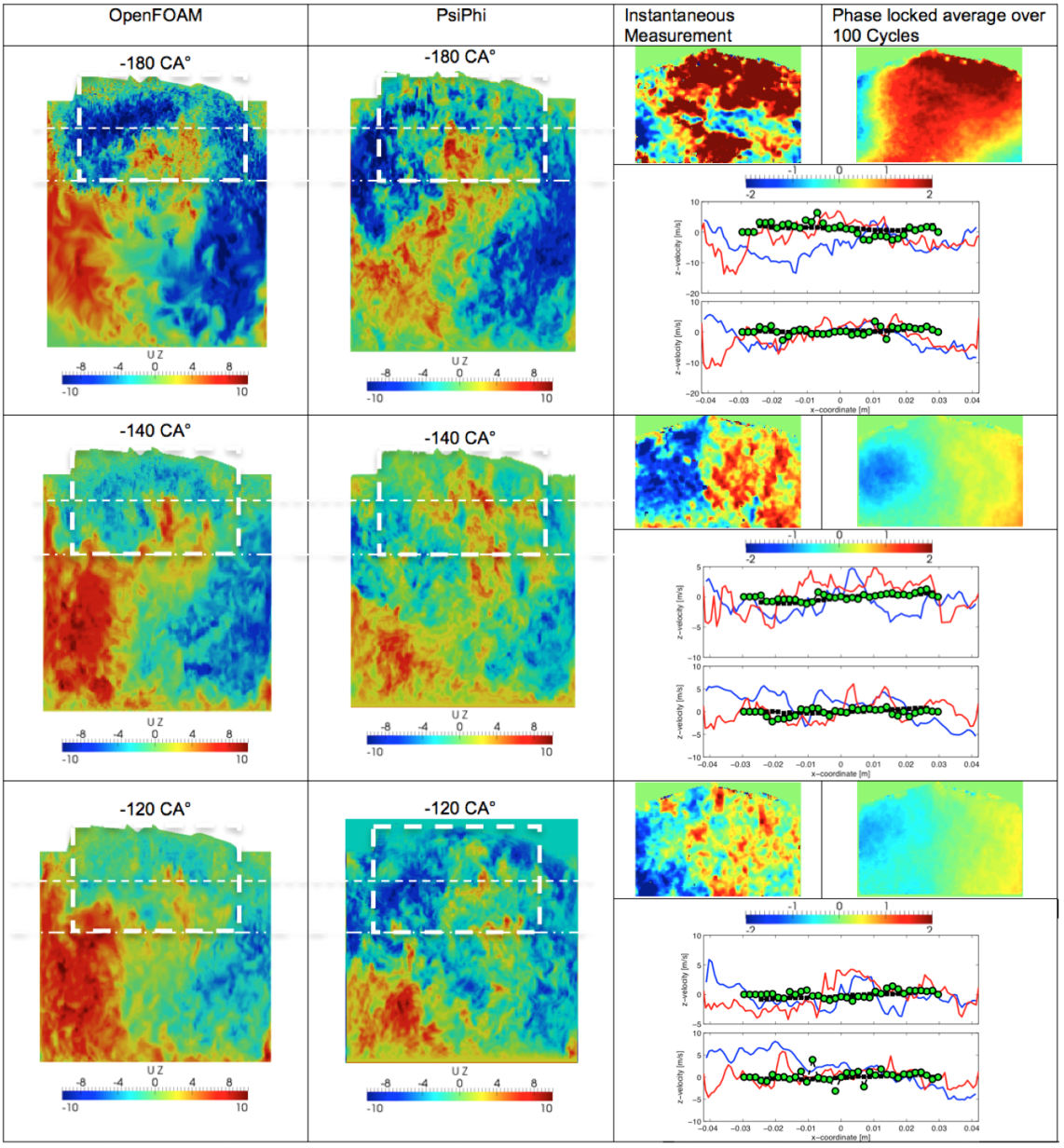

FIGURE 5.20: Contour plots of the z-velocity component of the tumble symmetry plane at -180 CA°, -140 CA° & -120 CA°. Velocity profiles for 10 mm(- -) and 30 mm (-···-) below the cylinder head. One measured instantaneous velocity contour plot is shown and the phase locked average from 100 cycles. (OpenFOAM (—○—), PsiPhi (—+—), instantaneous measurement (●) and phase-locked average over 100 cycles (■)).

Chapter 6

Large Eddy Simulation of an Internal Combustion Engine Using an Efficient Immersed Boundary Technique

Authors: T. Nguyen, F. Proch, I. Wlokas, A. Kempf

This chapter including all figures and tables was previously published in Flow, Turbulence and Combustion, Large Eddy Simulation of an Internal Combustion Engine Using an Efficient Immersed Boundary Technique. Flow, Turbulence Combustion Volume 97, 2016, pp 191-230, T. Nguyen, F. Proch, I. Wlokas, A. Kempf (©Springer Science+Business Media Dordrecht 2016) "With permission of Springer". The author T. Nguyen developed the code, ran all the simulations, wrote the paper and generated all figures and tables. The author F. Proch made important contributions in the development of the code. The original code was developed by A. Kempf. The author F. Proch, I. Wlokas and A. Kempf contributed corrections, discussions and proof-reading.

6.1 Abstract

This paper presents highly resolved large eddy simulations (LES) of an internal combustion engine (ICE) using an immersed boundary method (IBM), which can describe

moving and stationary boundaries in a simple and efficient manner. In this novel approach, the motion of the valves and the piston is modeled by Lagrangian particles, whilst the stationary parts of the engine are described by a computationally efficient IBM. The proposed mesh-free technique of boundary representation is simple for parallelization and suitable for high performance computing (HPC). To demonstrate the method, LES results are presented for the flow and the combustion in an internal combustion engine. The Favre-filtered Navier-Stokes equations are solved for a compressible flow employing a finite volume method on Cartesian grids. Non-reflecting boundary conditions are applied at the intake and the exhaust ports. Combustion is described using a flame surface density (FSD) model with an algebraic reaction rate closure. A simplified engine with a fixed axisymmetric valve (see Appendix A) is employed to show the correctness of the method while avoiding the uncertainties which may be induced by the complex engine geometry. Three test-cases using a real engine geometry are investigated on different grids to evaluate the impact of the cell size and the filter width. The simulation results are compared against the experimental data. A good overall agreement was found between the measurements and the simulation data. The presented method has particular advantages in the efficient generation of the grid, high resolution and low numerical dissipation throughout the domain and an excellent suitability for massively parallel simulations.

6.2 Introduction

Large eddy simulations are well suited for the investigation of the physical phenomena inside internal combustion engines with high demands for spatial and temporal accuracy [15]. Early applications of LES to non-premixed [178, 179, 180] and premixed [179, 181, 38] flames demonstrated the capability of LES in combustion modelling. The method has been extended and modified for the modelling of gas turbine combustors [182, 183, 184], spray combustion [185, 186, 187] and coal combustion [176, 188, 189].

In the past, LES was used for modelling of the gas exchange [190, 191] as well as the combustion process inside IC engines [16], utilizing a flame surface density model. Further LES investigations were performed to analyze the fuel-air mixing [192], knock tendency [57], turbulent heat transfer [193] as well as cyclic variations [194, 61].

The aim of the present work is to demonstrate the applicability of a novel approach based on immersed boundaries [164] for the LES of IC engines, which focuses on high numerical efficiency and parallel scalability on modern computer hardware architectures.

To generate meshes for complex geometries, Peskin [164] proposed a novel IBM based on Cartesian grids as a simple and efficient alternative to body-fitted grids. Based on Peskin's approach, different techniques have been developed for further improvement of the IBM in various applications. Numerous approaches of IBM can be categorized based on the numerical treatments of boundary forcing functions or based on the boundary types such as elastic or rigid. More information about IBMs can be found in the review paper of Mitall et al. [144], in which the methodologies, the advantages and the disadvantages of various IBMs are analyzed in a great detail. In this section, we focus our attention on the best known approaches. Goldstein et al. [195] introduced a surface body force as a function of velocity, time and two model parameters (α and β). This force represents the feedback force which is induced by the flow velocities on the immersed boundaries. In an approach of Verzicco et al. [196], suitable body forces are prescribed to yield the desired velocity on a given surface. The technique is simple, computational efficient, and flexible even in complex geometries.

A similar approach using an external force field was proposed by Balaras [197]. Utilizing the detailed topology description of the boundary to distribute the surface force properly, this methodology provides a highly accurate interface tracking scheme between fluid and solid regions and hence reduces the complexity related to the computation of the sub-grid fluxes in the solid interface's vicinity. This approach is well-suited for stationary immersed boundaries but lacks the proper treatment of moving boundaries where the interfaces between fluid and solid are a function of time and space. According to the Goldstein's assumption [195], the computational overhead is due to the increasing number of iterations for the desired constants α and β since the fluid-solid interface is time-dependent. The improved method of Balaras also introduces a large overhead which has been a standing problem for the community since a significant computational effort is required for the solid-fluid interface tracking with a second order accuracy scheme. Furthermore, as the B-splines method [198] is used for interface tracking, the moving boundary leads to grid restructuring, which makes the approach more expensive. Tullio et al. [199] developed an IBM for compressible flows using a flexible local refinement technique to resolve the flow near the immersed body. The method

is second-order accurate and can solve complex flows on a non-uniform Cartesian grid with a low computational cost compared to the corresponding fine equidistant Cartesian grid. However, applying this approach to the moving boundary would be problematic since the complicated grid with local refinement has to be regenerated frequently during the simulation. In the IBM proposed by Taira et al. [200], the pressure and boundary force are regarded as Lagrange multipliers required to maintain the divergence-free and the no-slip conditions. The boundary force is determined implicitly by a projection, in which the slip and non-divergence-free components of the velocity field are removed. This method is well-suited for both stationary and moving boundaries.

In the work of Udaykumar et al. [201], a mixed Eulerian-Lagrangian method is used to simulate the flow and geometries with complex and moving boundaries. Lagrangian particles are translated explicitly as a marker for the position and the shape of the boundaries, the governing equations are solved on an Eulerian grid. A straightforward method proposed by Mittal et al. [202] combines the cell blanking approach and the arbitrary Lagrangian-Eulerian method [203] by using two independent meshes: the stationary mesh and the moving mesh. The intersection between the moving grid and the fixed grid is calculated for the solid-fluid interfaces, allowing flexible mesh generation. However, dealing with mesh intersections between the fixed grid and the moving grid makes this method more complicated and computationally expensive. Forrer et al. [145] proposed the concept of 'mirror flow', in which the values of the cells inside the interior of a moving object are extrapolated from a given solution. This approach is developed to solve the compressible, time-dependent Euler equations on a Cartesian grid with stationary and moving walls. Our proposed IB technique in this paper is mainly based on this concept of 'mirror flow'. Since it was shown that the concept of 'mirror flow' is well-suited for hyperbolic problems [145], we applied it for the solution of the filtered compressible Navier-Stokes equations with moving boundaries.

In our approach, a structured grid is used throughout the whole computational domain in which the immersed boundary technique [164] is applied. Since the geometry description depends on the resolution of the Cartesian grid, the geometrical features which are smaller than a grid cell (Δ) can not be resolved. The geometry description on an uniform Cartesian grid is automatically generated from the STL representation of the geometry of the engine within minutes without further user interaction. To avoid the complexity of the mesh motion, a meshfree technique is applied to model the moving parts of

the simulated engine: the piston and the valves are represented by different groups of Lagrangian particles with corresponding motion functions. To be more precise, this technique employs the moving objects in two forms: Lagrangian particles and immersed boundaries. The Lagrangian form is utilized to handle the motion of the objects in the computational domain while the immersed boundary form is employed to impose the flow condition at the moving boundaries using the 'mirror flow' concept by Forrer et al. [145]. By blocking the computational cells that belong to the solid parts (where Lagrangian particles are located), no mesh alteration or new mesh is required during the simulation. The parallelization is highly efficient due to the fact that the domain decomposition is independent from the engine geometry.

The numerical and physical boundary conditions at the intake and exhaust ports are obtained by solving the system of equations for non-reflecting boundary conditions [136]. The combustion modelling in the power stroke is performed with a flame surface density model. In our opinion, an uniformly high resolution of the entire in-cylinder region is required (not just the near wall regions) to avoid artificial dissipation of the turbulent structures, which is a prerequisite to describe the flame front propagation accurately. Alternatively, adaptive grid refinement of the flame front may not circumvent well the issue of artificially dissipated turbulent structures and makes the computation rather complicated, expensive and possibly less reproducible. Figure 6.1 illustrates the instantaneous turbulent flow field and the fine structures that are resolved during an engine LES.

6.3 Modeling Approach

6.3.1 The governing equations

The filtered governing equations for mass (6.1), momentum (6.2), and energy (6.3) are outlined below.

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho} \widetilde{u}_j}{\partial x_j} = 0 \tag{6.1}$$

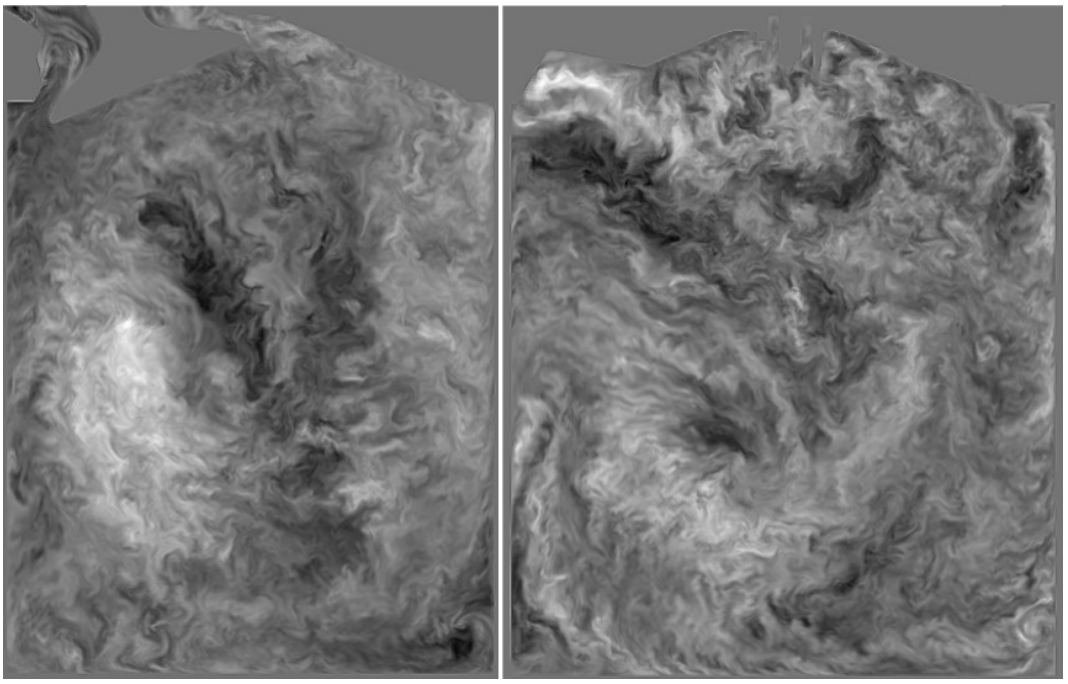

FIGURE 6.1: Instantaneous image-normal velocity component in the valve section (left) and in the tumble-symmetry plane (right) obtained from LES with a domain size of 180 million cells, each pixel corresponds to one computational cell with an edge length of 0.2 mm.

$$\begin{aligned} \frac{\partial \overline{\rho}\widetilde{u_i}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_j}\widetilde{u_i}}{\partial x_j} &= -\frac{\partial \overline{p}}{\partial x_i} \\ + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{\nu}\left(\frac{\partial \widetilde{u_i}}{\partial x_j} + \frac{\partial \widetilde{u_j}}{\partial x_i}\right) - \frac{2}{3}\overline{\rho}\widetilde{\nu}\frac{\partial \widetilde{u_k}}{\partial x_k}\delta_{ij} - \overline{\rho}\tau_{ij}^{sgs}\right] \end{aligned} \tag{6.2}$$

$$\begin{aligned} \frac{\partial \overline{\rho}\widetilde{e}}{\partial t} + \frac{\partial \left(\overline{\rho}\widetilde{e}\right)\widetilde{u_j}}{\partial x_j} &= \overline{\dot{w}}_T + \frac{\partial}{\partial x_j}\left(\kappa\frac{\partial \widetilde{T}}{\partial x_j}\right) \\ + \frac{\partial}{\partial x_j}\left(\widetilde{u_i}\widetilde{\sigma_{ij}}\right) - \frac{\partial}{\partial x_j}\left(\kappa_{ij}^{sgs}\right) \end{aligned} \tag{6.3}$$

In Eqs. (6.1-6.3), the density, velocity, pressure, molecular viscosity and Kronecker delta are denoted by $\overline{\rho}$, $\widetilde{u_i}$, $\overline{p}$, $\widetilde{\nu}$ and δ_{ij}, respectively. The unresolved stress and the unresolved kinetic energy are τ_{ij}^{sgs} and κ_{ij}^{sgs}. The total energy, Cauchy stress tensor and the temperature are represented by $\widetilde{e}$, $\widetilde{\sigma}_{ij}$ and $\widetilde{T}$, respectively.

The equation of state of an ideal gas is given in Eq. 6.4:

$$p = \rho R T \, , c_v = \frac{R}{\gamma - 1} \, , c_p = c_v + R \tag{6.4}$$

The gas constant, the heat capacity at constant volume and pressure are represented by R, c_v and c_p, respectively.

The total internal energy $\widetilde{e}$ and the sensible internal energy $\widetilde{e}_s$ are determined according to Eq. 6.5 (Poinsot and Veynante [71]).

$$\overline{\rho}\widetilde{e} = \frac{1}{2}\overline{\rho}\widetilde{u}_k\widetilde{u}_k + \overline{\rho}\widetilde{e}_s \,, \quad \widetilde{e}_s = \int_{T_0}^{T} c_v(T)\, dT \tag{6.5}$$

The combustion modelling of this study is discussed in section 6.3.6. For the investigated test cases, the air and the fuel are assumed to be perfectly mixed and hence no additional transport equation is needed to describe the composition of the mixture. The heat release due to combustion is computed by $\overline{\dot{w}}_T = \sum_{k=1}^{n} \Delta h^o_{f,k}\dot{\omega}_k$ as the product of the formation enthalpy $\Delta h^o_{f,k}$ of species k and the reaction rate $\dot{\omega}_k$.

6.3.2 From moving boundaries to Lagrangian particles

The moving boundaries are numerically described by Lagrangian particles, which must be generated prior to the simulation using available software for "voxelization" (voxel: volume element). In our approach, the voxelization step must be carried out first to convert a geometric object (the moving object) into a set of voxels with the resolution of the computational domain. Obviously the number of voxels describing any moving geometries depends on the resolution of the computational grid. This set of voxels is then mapped onto the main computational grid for correct positioning. By then, the indices of each voxel in the computational domain will be assigned to a Lagrangian particle's index and the initial coordinate of a particle can be computed from its indices. Figure 6.2 illustrates the basic steps to generate Lagrangian particles from the original geometry. By using Lagrangian particles for the description of the moving parts, the motion of these objects is characterized as the motion of the individual particles in a Lagrangian manner. Therefore, the transport equations for the individual particles in space and time must be considered. Generally, based on the particle velocity v_{pi} and the location x_i^{old} from the last time step, the new location x_i^{new} is calculated according to Eq. 6.6.

$$x_i^{\text{new}} = x_i^{\text{old}} + v_{pi}^{\text{old}}\Delta t \tag{6.6}$$

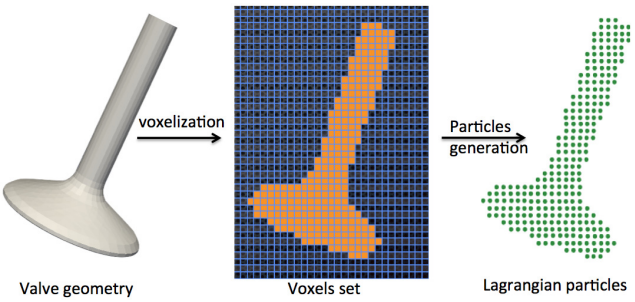

FIGURE 6.2: Two-step procedure to generate Lagrangian particles from the original geometry. First, voxelization converts a geometric object into a set of voxels, then, a particle is created from each voxel.

The motion of the piston, the intake and exhaust valves are determined as a function of the crank angle. From the valve lift profiles, the velocity v_{pi} and acceleration a_{pi} of the group of the particles describing the intake or the exhaust valves are calculated from Eqs. 6.7-6.8.

$$v_{pi} = \frac{x_i^{\text{new}} - x_i^{\text{old}}}{\Delta t} \tag{6.7}$$

$$a_{pi} = \frac{v_{pi}^{\text{new}} - v_{pi}^{\text{old}}}{\Delta t} \tag{6.8}$$

In contrast to the valve particles, the motion of the piston particles is modeled in a simpler manner since the piston velocity and the piston acceleration can be analytically calculated from the piston kinematics based on the crank angle (CA) Θ, the conrod length l, the crank radius r, and from the angular velocity ω ($\omega = 2\,\pi$ rpm / 60) from Eqs. 6.9-6.10 [106]. These simplified equations (no offset of the piston) are sufficient since there is no such offset in the experiment.

$$v_{pi} = \left[-r\sin\Theta - \frac{r^2 \sin\Theta\cos\Theta}{\sqrt{l^2 - r^2\sin^2\Theta}} \right]\omega \tag{6.9}$$

$$a_{pi} = \left[-r\cos\Theta - \frac{r^2(\cos^2\Theta - \sin^2\Theta)}{\sqrt{l^2 - r^2\sin^2\Theta}} - \frac{r^4(\cos^2\Theta\sin^2\Theta)}{\sqrt{l^2 - r^2\sin^2\Theta}} \right]\omega \tag{6.10}$$

Besides the actual particle positions x_i, the logical particle coordinates (i,j,k) within the grid must be computed from the actual position x_i and the cell size Δ, considering

nG layers of ghost cells that may be used around the computational domain. For the isotropic and (i,j,k) countable computational grid (Cartesian grids) used in our approach, this is achieved by following Eq. 6.11.

$$i = \left\lfloor \frac{x_i}{\Delta} \right\rfloor + 1 + nG \tag{6.11}$$

After transporting the particles to the new position in the computational domain, the cell-blocking procedure is performed to establish the new interface between fluid and solid regions. This step is executed by introducing an immersed boundary field (IBF) as the partition of fluid and solid zones. Taking full advantage of the equidistant Cartesian grid, the blocking of solid cells can be done simply by using the calculated indices i,j,k of the corresponding cell where a particle is located:

$$\mathrm{IBF}_{i,j,k} = 0 \tag{6.12}$$

6.3.3 Impose boundary conditions at the moving boundaries

In order to include the effect of the moving boundaries on the surrounding fluid, most of the aforementioned approaches (see section 6.2) introduced a forcing function $\boldsymbol{f_b}$ and then modified the equation system $L(\boldsymbol{\Phi}) = 0$ to the new one $L(\boldsymbol{\Phi}) = \boldsymbol{f_b}$, accordingly ($\boldsymbol{\Phi}$ is the vector of conservative variables). The forcing function can be included into the governing equations before (continuous forcing approach) or after (discrete forcing approach) discretization [144]. An iterative calculation must be carried out to obtain this forcing function for the flow around a rigid body.

Instead of introducing a forcing function $\boldsymbol{f_b}$, in our approach, the effect of the moving boundaries Γ_b on the surrounding fluid is included via the 'mirror flow' concept [145]. The discretized governing equations are modified by imposing the 'mirror flow' quantities $\boldsymbol{\Phi_\Gamma}$ inside the immersed boundaries Γ_b . These 'mirror' quantities $\boldsymbol{\Phi_\Gamma}$ are extrapolated from neighboring fluid cells and from the particle properties, so that the flow conditions at the moving boundaries are imposed correctly.

The procedure is as follows: after solving the discretized Navier-Stokes equations $L(\boldsymbol{\Phi}) = \mathbf{0}$ for the numerical solutions $\boldsymbol{\Phi^n}$ at time step t_n, the 'mirror flow' quantities $\boldsymbol{\Phi_\Gamma}$ of the solid cells inside the moving boundaries Γ_b are calculated based on the predetermined

particle's properties (v_{pi}^{n+1} and a_{pi}^{n+1}) at time step t_{n+1} and the data extrapolated from the neighboring fluid cells $\mathbf{\Phi^n}$. Afterward, the discretized Navier-Stokes equations are solved for the numerical solution $\mathbf{\Phi^{n+1}}$ in the whole computational domain. In other words, the explicit time-stepping is performed with the moving boundary positions from the time step t_{n+1} before advancing the whole numerical solution to the next time step t_{n+1}.

The imposed velocities v_{si} of the solid cells at the solid-fluid interfaces are computed from the particle's velocity v_{pi} and the velocity v_{fi} of the neighboring fluid cells N_{nb}:

$$v_{si} = 2v_{pi} - \frac{\sum_{j=1}^{N_{nb}} v_{fij}}{N_{nb}} \tag{6.13}$$

Following Forrer et al. [145], the pressure gradient $\partial p/\partial x_i$ of a boundary cell is calculated from the corresponding particle's acceleration a_{pi} and the averaged density ρ_j of the neighboring fluid cells:

$$\frac{\partial p}{\partial x_i} = -\frac{\sum_{j=1}^{N_{nb}} \rho_j}{N_{nb}} a_{pi} \tag{6.14}$$

To be more specific, the numerical quantities of a solid cell at the boundary require averaged quantities from the neighboring fluid cells N_{nb} where $0 \leq N_{nb} \leq 6$.

Since the pressure p and the velocities of the solid cells v_{si} are set, the corresponding energy e_{si} is also required to account for the kinetic energy of the particles' motion as well as the sensible energy of the cylinder wall:

$$e_{si} = \frac{1}{2} v_{si}^2 + c_v T_{si} \tag{6.15}$$

The heat flux at the wall boundary is calculated by Fourier's law $q_i = -\kappa \frac{\partial T}{\partial x_i}$ with the gas thermal conductivity κ. The isothermal boundary condition is used by setting the wall temperature to a fixed value. Due to the lack of experimental data, the temperature at the moving boundaries is set to the same value as the wall temperature T_{wall} of the cylinder. In the current work, we do not use any additional modelling of wall heat transfer - a known and difficult problem for engine LES.

6.3.4 Closing and opening the valves

The valve gap during simulation can only be resolved up to a certain valve lift value depending on the grid resolution. In order to close the valves, all the computational cells in the gap (the smallest resolved valve lift) between the valves and the wall are blocked ($\mathrm{IBF}_{ijk} = 0$), so that preventing the fluid escaping out of the cylinder.

In order to reopen the valves, the blocked cells in the gap between the valve and the wall are unblocked (the IBF value is changed from 0 to 1). Since these freshly cleared cells (the cells, which were solid at time step t_n, become the fluid cells at time step t_{n+1}) have no fluid history, it is important to fill these cells with the flow data (velocity, density, temperature and pressure) by interpolating the data from neighboring fluid cells. Alternatively, a simple one-dimensional interpolation operation proposed by Udaykumar et al. [204] can be used to fill the freshly cleared cells with the flow data.

6.3.5 The intake and the exhaust boundary conditions

Based on the measured mass flow and the measured time-dependent pressure, non-reflecting boundary conditions are enforced at the end of the truncated inlet and exhaust ports, which have a length of 60 and 110 mm, respectively. Depending on the amplitude variations of characteristic waves which are entering or leaving the computational domain, the time-dependent physical flow conditions are computed by solving the locally one-dimensional and inviscid (LODI) system as proposed by Poinsot et al. [136]. The obtained values for the density ρ, velocities u_i, pressure p and the energy e are imposed on the cells at the inlet and the outlet of the domain.

6.3.6 Combustion modelling

The combustion process is modeled with the flame surface density (FSD) approach. The flame propagation is described by the progress variable c, which is based on the fuel mass fraction Y_F, where Y_F^u denotes the unburned and Y_F^b the burned state ($Y_F^u = 0$ for lean condition).

$$c = \frac{Y_F - Y_F^u}{Y_F^b - Y_F^u} \tag{6.16}$$

The transport equation for the Favre-filtered progress variable reads:

$$\begin{aligned}&\frac{\partial\overline{\rho}\widetilde{c}}{\partial t}+\frac{\partial\overline{\rho}\widetilde{c}\widetilde{u}_i}{\partial x_i}+\frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu}_i-\widetilde{c}\,\widetilde{u}_i)]\\&=\overline{\frac{\partial}{\partial x_i}\left(\rho D\frac{\partial c}{\partial x_i}\right)}+\overline{\dot{w}}_c=\overline{\rho S_d}\Sigma_{\text{gen}}\end{aligned}\tag{6.17}$$

In this equation, the mean reaction source term and the molecular diffusivity are $\overline{\dot{w}}_c$ and D. The displacement speed and the generalized flame surface density are denoted as S_d and Σ_{gen}, where the generalised flame surface density combines molecular diffusion and the source term.

The sub-grid scalar flux and the generalised flame surface density are modeled by Eqs. 7.2-7.3.

$$\frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu}_i-\widetilde{c}\widetilde{u}_i)]=\frac{\partial}{\partial x_i}\left(\overline{\rho}\frac{\nu_t}{S_c}\frac{\partial\widetilde{c}}{\partial x_i}\right)+F_{cgt}\tag{6.18}$$

$$\begin{aligned}\overline{\rho S_d}\Sigma_{\text{gen}}\approx\rho_u s_L\Sigma_{\text{gen}}&=\rho_u s_L\Xi|\nabla\overline{c}|-F_{cgt}\\&\approx\rho_u s_L\Xi|\nabla\widetilde{c}|\end{aligned}\tag{6.19}$$

Comment: the counter gradient term F_{cgt} is presented in section 2.7 of Chapter 2.

In these equations, the turbulent viscosity, the turbulent Schmidt number, the laminar flame speed and the wrinkling factor are referred to as ν_t, Sc, s_L and Ξ, respectively. As it has been shown by Ma et al. [103], the counter gradient transport F_{cgt} is implicitly modeled by replacing the Reynolds-averaged by the Favre-filtered progress variable in Eq. 7.3. The unburned gas density ρ_u is computed from the ambient pressure and density (p_{in}, ρ_{in}) as given in Table 7.1, the heat capacity ratio γ and the actual pressure inside the engine according to Eq. 6.20.

$$\rho_u=\rho_0\left(p/p_0\right)^{(1/\gamma)}\tag{6.20}$$

The wrinkling factor is evaluated from the model proposed by Muppala et al. [32] according to Eq. 7.4, which is preferred here since it considers the effect of the pressure variation on the flame propagation which is important for engine simulations.

$$\Xi=1+a\text{Re}_t^{0.25}\left(\frac{u'}{s_L}\right)^b\left(\frac{p}{p_0}\right)^c\tag{6.21}$$

The turbulent Reynolds number Re_t is computed as $\mathrm{Re}_t = u'\Delta/\nu$ with the cell size Δ and the laminar viscosity ν. For the iso-octane/air flame, the model parameters are set to $b = 0.2$, $c = 0.2$, and $a = 0.46/\mathrm{Le}$ [32, 106].
The sub-grid scale velocity fluctuation u' is estimated according to Wyngaard [107]:

$$u' = \sqrt{\frac{2}{3}\kappa_{sgs}}, \quad \kappa_{sgs} = \frac{1}{(C_v\Delta)}\nu_t^2 \tag{6.22}$$

We chose $C_v = 0.1$ [108] and ν_t is obtained from the Nicoud's σ model [87].

The laminar flame speed can be determined according to Heywood [106]:

$$s_L = s_{L,0}\left(\frac{T_u}{T_0}\right)^{\alpha}\left(\frac{p}{p_0}\right)^{\beta} \tag{6.23}$$

In Eq. 6.23, the unburned gas temperature T_u is calculated from $T_u = T_0\left(p/p_0\right)^{\frac{\gamma-1}{\gamma}}$. The parameters $s_{L,0}$, α and β depend on the specific fuel, the equivalence ratio and the burned gas dilution fraction [106]. Wall heat losses were considered to be distributed homogeneously over the combustion chamber by a modified heat capacity ratio of $\gamma = 1.3$. This treatment was necessary as a detailed heat transfer modelling would have required an even finer grid resolution or a suitable wall model.

The ignition is created by setting the progress variable c in a small number of cells near the spark-plug to a value of 1.0 (burned area). The minimum ignition energy that is added to the system and the critical radius of the initial flame kernel are estimated based on the flammability limits [113].

6.4 Experiments

The measurements were performed in a four-stroke optical engine operated in the Dreizler group at Darmstadt University [141], using iso-octane (C_8H_{18}) as the surrogate fuel. Optical access is provided by a quartz glass cylinder liner and a flat piston window. The engine features a twin-cam, an overhead-valve pent-roof cylinder head with two intakes and two exhaust valves. A detailed description can be found in Tables 6.1, 7.1 and in the paper by Baum et al. [141].

Particle imaging velocimetry (PIV) was applied to capture the flow field within the cylinder. The spatial resolution of the data shown in this paper was 1.8 mm. The engine is motored at 800 rpm and 1500 rpm with a geometric compression ratio of 8.5:1. The measurements were carried out over 2700 and 600 cycles for the cases with 800 and 1500 rpm, respectively. The measurements showing the flame propagation (Fig. 6.18) were provided by Brian Peterson from Darmstadt University (private communication). High-speed imaging of Mie-scattering of evaporating oil droplets within the reaction zone was used to distinguish between burned and unburned gas regions within a horizontal swirl plane 1.3 mm below the spark plug, similar to the procedure presented in Fansler et al. [205].

Data	Value
Engine speed	800 rpm & 1500 rpm
Compression ratio	8.5
Displacement volume	499 cm^3
Crevice volume	2 cm^3
Volume at top dead center (TDC)	66.5 cm^3
Bore	86 mm
Stroke	86 mm
Cylinder clearance height	2.6 mm
Intake valve closure (IVC)	-125 CA
Exhaust valve open (EVO)	105 CA
Exhaust valve close (EVC)	-345 CA
Intake valve open (IVO)	325 CA
Avg. pressure intake (p_{in})	0.95±0.02 bar
Avg. pressure exhaust (p_{out})	1.00±0.016 bar
Intake temperature (T_{in})	22.9±0.1°C
Exhaust temperature (T_{out})	43.7±0.1°C
Mass air in $< m_{in} >$	11.5 kg/h±2 %
Mass air out $< m_{out} >$	11.5 kg/h±2 %

TABLE 6.1: Engine specifications in the motored case.

6.5 Numerical Setup

Simulations were performed on three different grids with cell sizes of 0.8 mm, 0.5 mm and 0.2 mm, as illustrated in Fig. 6.3. This leads to domains of 3.1, 12.7, and 180 million cells. The computations were performed on 24, 240 and 8196 CPUs, respectively. The ratios between the number of solid cells and the number of fluid cells for different grid sizes when the piston is located at the TDC/BDC are shown in Table 6.3.

Data	Value
Engine speed	800 rpm
Compression ratio	8.5
Spark timing (ST)	-16 CA
Fuel	Iso-Octane
Equivalence ratio (ϕ)	0.833
Avg. pressure intake (p_{in})	0.95±0.02 bar
Intake temperature (T_{in})	47±3°C
Intake density (ρ_{in})	1.0344 kg/m^3

TABLE 6.2: Engine specifications in the combustion case.

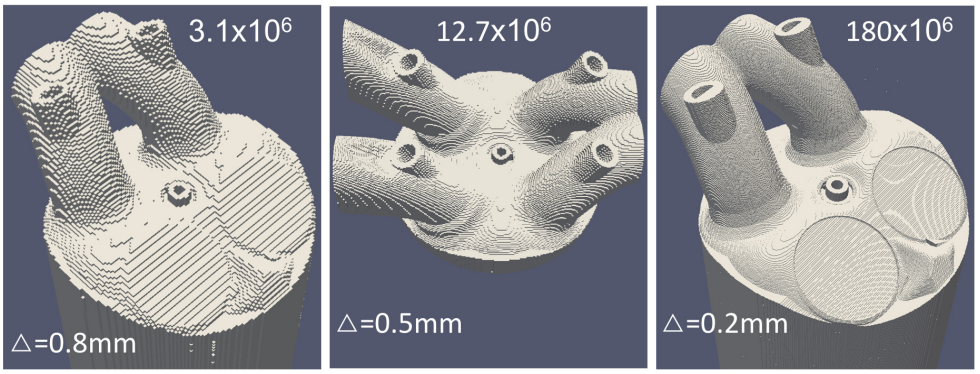

FIGURE 6.3: The Cartesian grids of the engine geometry in different resolutions (Δ = 0.8 mm, 0.5 mm, and 0.2 mm).

Grid	BDC	TDC
Δ = 0.8 mm	$1.74\times10^6/1.36\times10^6$	$2.26\times10^6/0.84\times10^6$
Δ = 0.5 mm	$7.0\times10^6/5.7\times10^6$	$9.2\times10^6/3.5\times10^6$
Δ = 0.2 mm	$95.4\times10^6/84.6\times10^6$	$124.6\times10^6/55.4\times10^6$

TABLE 6.3: The ratio of the number of the solid cells/fluid cells at TDC and BDC.

The LES in-house code PsiPhi [162, 157, 173, 36, 187] was used to solve the Favre-filtered governing equations for a compressible fluid (Eqs. 6.1-6.3). The code uses simple equidistant grids, thus requires more cells for a simulation than an non-uniform grid would, which is often (over)-compensated by the better accuracy and the fact that the computational costs per cell are, in our experience, one to two orders of magnitude lower. A clear advantage of our approach is that the numerical error corresponds exactly to the error of the applied interpolation scheme, while on irregular grids additional terms may appear and contribute to the truncation error. Furthermore, schemes of higher order require on unstructured grids very complicated and large stencils - which is computationally expensive (even though unstructured codes like AVBP [206] and YALES2 [207] demonstrate that 3^{rd} order schemes are practicable).

Time integration is performed with an explicit third order Runge-Kutta scheme. Due to temporal accuracy requirements and the limited stability of the explicit time integration scheme, the time step was constrained by the magnitude of the flow velocity $|u_i|$, the speed of sound a and a CFL number of 0.3 according to:

$$\Delta t = 0.3 \frac{\Delta}{|u_i| + a} \tag{6.24}$$

A second order central differencing scheme is applied for the computation of the momentum fluxes. For density and scalars a total variation diminishing (TVD) scheme with a nonlinear CHARM limiter [120] is used. The unresolved turbulent viscosity is determined with Nicoud's σ model [87, 208], where the model constant C_m is set to 1.5.

In our method, the numerical operations that solve the ordinary differential equations (ODEs) for the particle transport, block the computational cells (where the particles are located), and enforce the boundary conditions at the moving walls can be performed with a high computational efficiency on parallel machines. The code utilizes the optimized built-in Fortran vectorized field operations on a Cartesian grid. Scalability tests (in Table 6.4) of PsiPhi on the Cray-XT6m at the University of Duisburg-Essen (Fig. 6.4) confirm the capability of the approach. In these tests, the cost of transporting particles and enforcing the flow conditions at the moving boundaries is less than 5% although the particles are not equally distributed across parallel domains. Obviously, there is room for improved efficiency but this would affect the simplicity and possibly the speed of the overall simulation: it is not clear if an optimized domain decomposition and particle distribution, with all the resulting overhead, would be really beneficial.

Test case	CPUs (Cartesian topology)	Cells per CPU	Nparticles/Ncells
1	560 (10x8x7)	1784000	0.2%
2	960 (12x10x8)	1047230	0.2%
3	1440 (15x12x8)	694330	0.2%
4	1920 (16x12x10)	522210	0.2%
5	2400 (20x12x10)	415740	0.2%
6	3120 (20x13x12)	319800	0.2%
7	4080 (20x17x12)	243984	0.2%

TABLE 6.4: Configurations for scalability test case with PsiPhi.

In order to test the performance of our approach in a massively parallel simulation, a test case using more than 180 million cells was computed on the Blue Gene/Q machine

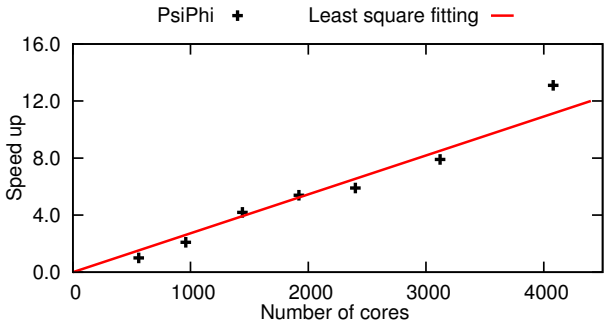

FIGURE 6.4: Strong scaling test on a Cray-XT6m.

JUQUEEN (FZ Juelich through NIC Germany) using 8196 cores. The cost of all the necessary steps (voxelization & particle generation) to produce the input geometry for the simulation was less than one cpu-h. The total number of particles (including intake, exhaust and piston particles) was less than three million even on the finest grid. Although solving the ordinary differential equations must be carried out for all the particles, the operators to enforce the flow conditions at the interface between fluid and solid are only performed using the boundary particles which usually account for 15-30 % of the total particles. Therefore, the total number of numerical operations to handle the moving boundaries in our proposed approach is relatively small, compared to the numerical treatment that is required for solving the Navier-Stokes equation. On the finest grid with 180 million cells and a resolution of 0.2 mm throughout the entire computational domain, to complete the simulation of one full cycle, the total computational cost was around 3.15 millions cpu-h for the motored-case at 1500 rpm.

6.6 Results

6.6.1 Motored-case at 800 rpm

Figure 6.5 shows a comparison between the velocity magnitude obtained from the simulation on the 0.5 mm grid and the PIV measurements at -270° CA. A fair quantitative agreement between the measured and the simulated velocity structures can be observed. The unsteady nature of the flow field becomes clearly visible as the instantaneous velocity profiles from the medium simulation are plotted along with the mean values in Figs. 6.6, 6.7, 6.12 and 6.13.

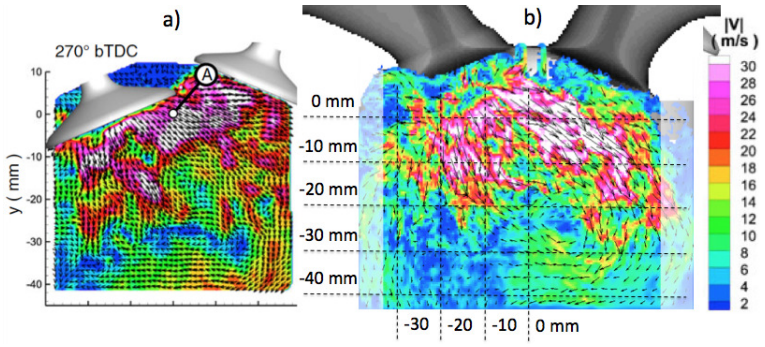

FIGURE 6.5: Illustration of the instantaneous measured velocity magnitude (a) at -270° CA (Reprinted from Baum et al. [141] with permission from Springer) and the corresponding velocity magnitude° obtained on the medium grid (b) with all the sampling lines.

Figures 6.6 and 6.7 compare the measured phase averaged velocity against the mean velocities obtained on different grids and 15 instantaneous velocity profiles from the medium grid at the sampling lines, shown in Fig. 6.5, for a crank angle of -270° CA. Comparisons are shown for the medium and coarse grid, averaged over 15 cycles. Good agreement was observed for all investigated grid resolutions. Even on the coarsest grid (0.8 mm) the flow dynamics could be represented correctly. The comparison at -270° CA in Fig. 6.6 and Fig. 6.7 indicates good agreement between the simulations and the experimental data for all the sampling lines, although the horizontal velocity profiles (u) at the lines -30 mm and -40 mm below the cylinder head (Fig. 6.6) show some deviation from the corresponding measurements. The differences of the horizontal velocity (u) at the aforementioned sampling lines could be likely anticipated since the kinetic energy of the intake flow is strongly dissipated in the downstream region of the cylinder and therefore leads to the relatively small velocities in this area. In the downstream region, where the horizontal velocity component is usually small, a minor disturbance created during the intake could significantly affect the flow velocity in the horizontal direction. Moreover, since the downstream region is close to the piston surface, the boundary layer, which is not well-resolved, may also lead to the deviation between the simulated and the measured data. Along the tumble plane of the cylinder, good agreement between the numerical results and the measurements is achieved for both the coarse (0.8 mm) and the medium (0.5 mm) grid as shown in Fig. 6.7. For the vertical sampling lines, small discrepancies between the horizontal velocity component (u) and the corresponding measured velocity are also found in the downstream region. The similar flow pattern with the distinctive flow structures and vortices is illustrated in Fig. 6.8, where a good

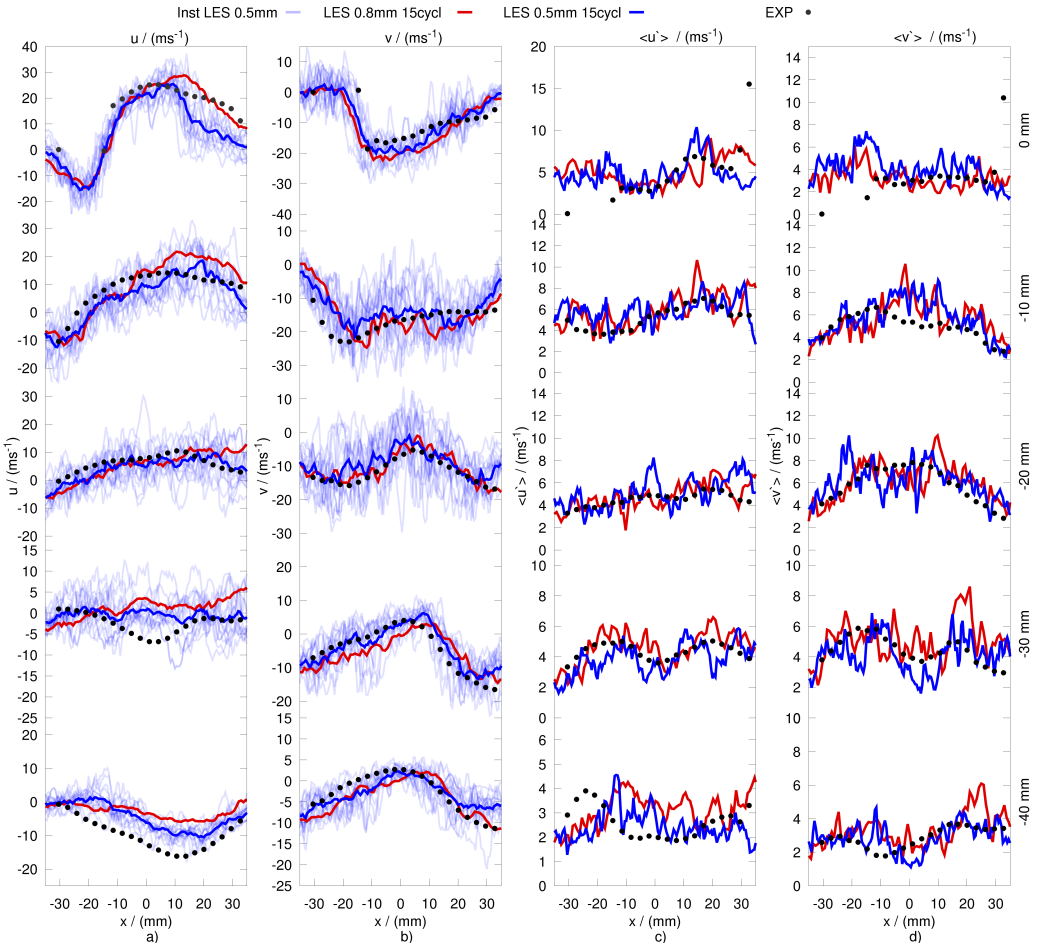

FIGURE 6.6: Predicted and measured velocity profiles at -270° CA. Lines in sub-figures (a) and (b) represent the phase averaged horizontal velocity component u and the vertical velocity component v. Additionally, 15 instantaneous velocity profiles obtained by the simulation on the medium grid are shown. Sub-figures (c) and (d) illustrate the corresponding RMS of u and v, respectively. Experimental results are shown with symbols. Results are shown for the horizontal sampling lines at 0 mm, -10 mm, -20 mm, -30 mm and -40 mm (see Fig. 6.5).

agreement between the simulations and the measurement is achieved. Despite the coarse grid, the results obtained from the simulation ($\Delta = 0.8$ mm) show a strong resemblance to the measurement since the larger number of engine cycles are converged towards the ensemble-mean flow. A good agreement between the measured and simulated root mean square (RMS) for the two velocity components (u,v) is also shown in Figs. 6.6, 6.7, and 6.9 for -270° CA. The RMSs over 15 cycles predicted by the simulations with the coarse and the medium grids show the clear trend of convergence towards the mean flow.

Capturing the correct in-cylinder flow during the intake stroke is essential since the turbulence is generated mainly in this stage and then later affects the flame propagation

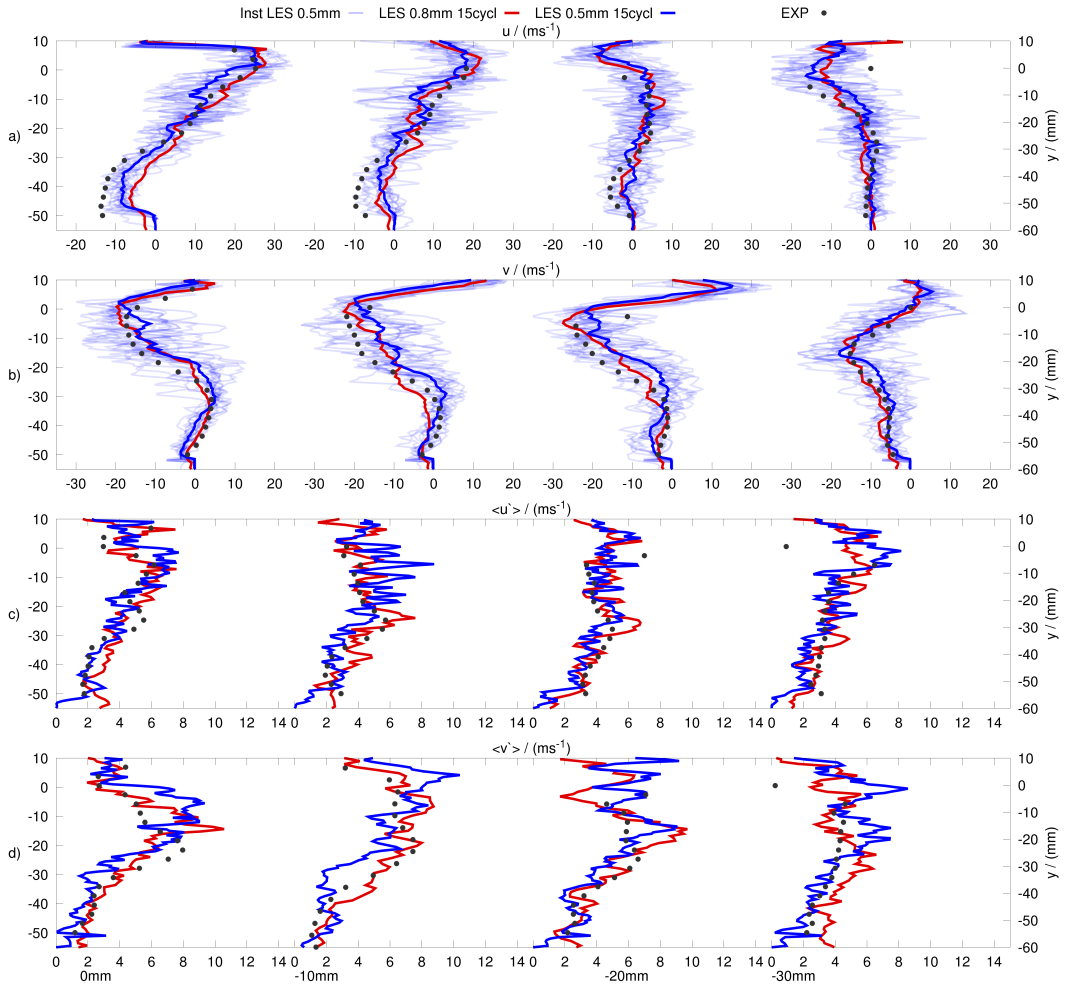

FIGURE 6.7: Predicted and measured velocity profiles at -270° CA. Lines in sub-figures (a) and (b) represent the phase averaged horizontal velocity component u and the vertical velocity component v. Additionally, 15 instantaneous velocity profiles obtained by the simulation on the medium grid are shown. Sub-figures (c) and (d) illustrate the corresponding RMS of u and v, respectively. Experimental results are shown with symbols. Results are shown for the vertical sampling lines at 0 mm, -10 mm, -20 mm and -30 mm (see Fig. 6.5).

during combustion. However, the valve-overlap in the period between the end of the exhaust stroke and the beginning of the intake stroke creates complicated flow dynamics since the fresh gas and the exhaust gas enter through the intake and the exhaust valves respectively, and interfere with each other. The good agreement found in the comparison between the numerical results and the measurement at -270° CA demonstrates the capability of our proposed approach, with which the complex flow behavior can be well-captured.

Agreement is also achieved for the comparison of the velocity profiles between the simulations and the measurements at -90° CA during the compression stroke as shown in

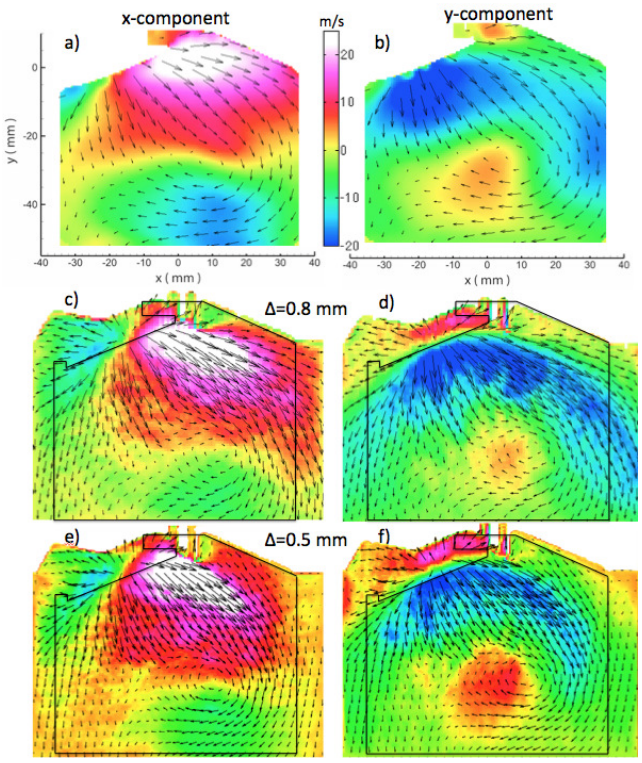

FIGURE 6.8: Illustration of the phase-averaged velocity components as obtained by the measurement (a,b) and the simulation on the coarse (c,d) and the medium grid (e,f) at -270° CA.

Figs. 6.10, 6.12 and 6.13.

An overall agreement is achieved at all sampling lines since the flow velocities in the tumble plane are well-captured. Unlike the flow dynamics, which is characterized by high velocity flow due to the pressure difference during the intake stroke, the in-cylinder flow during the compression stroke is driven by the piston motion and the inertia in the flow. Undergoing recirculation, interactions between tumble, eddies, moving walls, and the dissipation of kinetic energy make the dynamics of the flow under compression difficult to predict. Therefore capturing the correct velocity profiles at -90° CA is a challenging task. Although the numerical results obtained do not fully match the measurements at -90° CA, the flow velocities are well-predicted for most parts. Unlike the RMSs at -270° CA, RMSs at -90° CA show large uncertainty areas in the numerical results (Figs. 6.11c-6.11f) compared to the measurements (Figs. 6.11a-6.11b). Possibly, the flow behavior during compression (-90° CA) is more sensitive and requires a large number of cycles for converging towards the ensemble mean flow. Another possible reason could be the boundary layer, which is not sufficiently resolved in the near-wall regions. However,

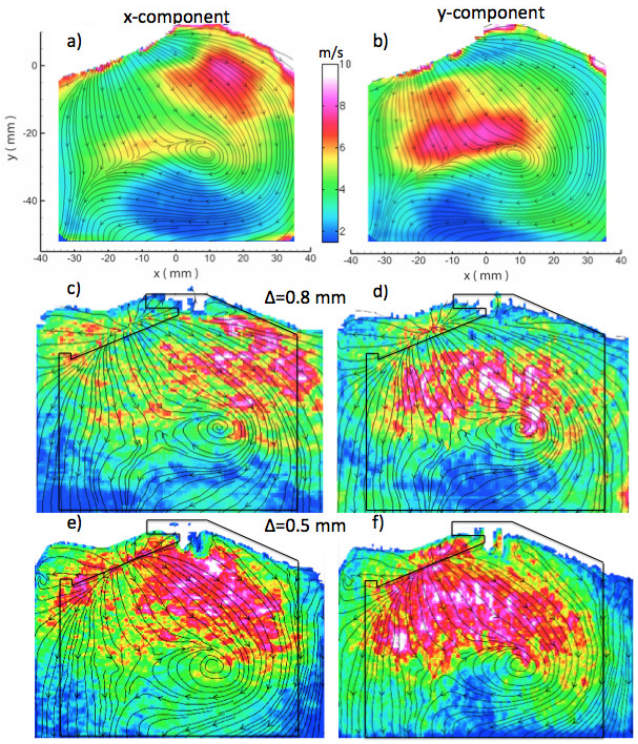

FIGURE 6.9: The standard deviation for the two measured velocity components at -270° CA (a,b). The corresponding RMS of the coarse simulation (c,d) ($\Delta = 0.8$ mm). The corresponding RMS of the medium simulation (e,f) ($\Delta = 0.5$ mm).

according to our experience, the σ model behaves well near walls. Nevertheless, the deviations observed by our simulations are similar to those in the predictions by Baumann et al. [158] where local refinement and wall modelling were employed to resolve the flow near the walls albeit at a lower grid resolution than used by us.

To estimate the convergence of the approach, the RMSs of two velocity components (x and y) at four specific points (p1(0,-40), p2(0,-10), p3(30,-15), and p4(-7.5,-17.5) mm) in the tumble symmetry plane are plotted as a function of the number of engine cycles. These points are chosen to be consistent with the measurements [141]. As seen in Fig. 6.14, the convergence rate for the predicted RMS of the coarse and the medium grid is quite visible after 15 simulated cycles.

The in-cylinder pressure curves during intake, compression, expansion and exhaust are compared to the measurements in Fig. 6.16a. The simulated pressure curves match the measurements well for the motored-case. At TDC, the simulated pressure is 0.5 bar lower than the measured one. This deviation may occur due to the fact that the wall temperature was not measured during the experiments and was assumed to be 400 K in

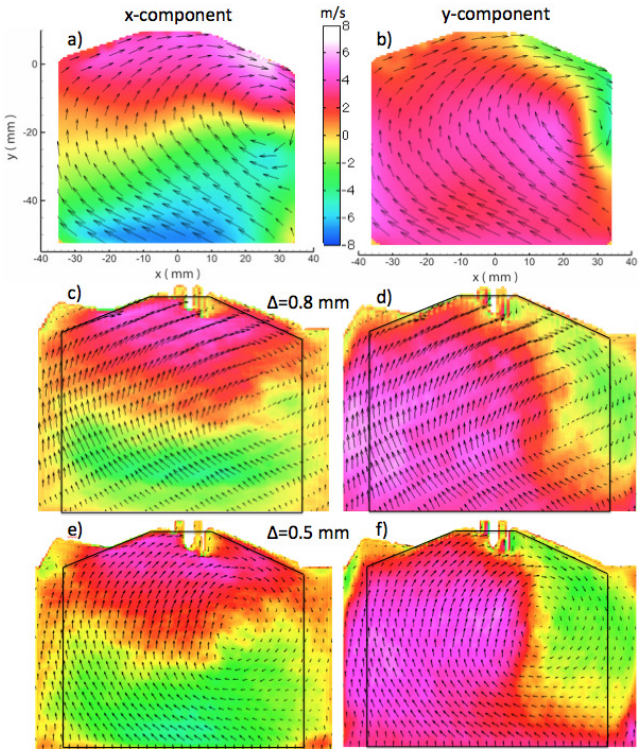

FIGURE 6.10: Illustration of the phase-averaged velocity components as obtained by the measurement (a,b) and the simulation on the coarse (c,d) and the medium grid (e,f) at -90° CA.

the simulations. Another reason for the low in-cylinder pressure may be the unresolved valve gap (a situation where the valve gap is smaller than one filter or cell size) which allows more gas to escape from the cylinder before IVC. To demonstrate that the mass is conserved inside the cylinder, the trapped mass during compression and expansion is plotted in Fig. 6.16a*.

The ratio of turbulent to laminar viscosity may be used as an indicator for regions where the grid is too coarse. Inside the cylinder (Fig. 6.15), this ratio mostly remains smaller than 10 (Fig. 6.15a, 6.15b and 6.15c) for the medium grid, thus highlighting a good quality of the medium grid [209]. The turbulent to laminar viscosity ratio exceeds the value of 20 in a small area (Fig. 6.15d) for the coarse grid. This ratio also shows the thin dissipation layers that are resolved and a high resolution achieved by the LES. It should be stressed that the velocities of the in-cylinder flow reach their maximal values at -270° CA, which subsequently leads to a large value of the viscosity ratio. In a complete engine cycle, the viscosity ratio for the coarse grid is mostly below a value of 5 (Fig. 6.15b, Fig. 6.15c, Fig. 6.15e, and Fig. 6.15f), indicating the high quality of even the coarse grid. It should also be noted that comparisons for two different grid resolutions

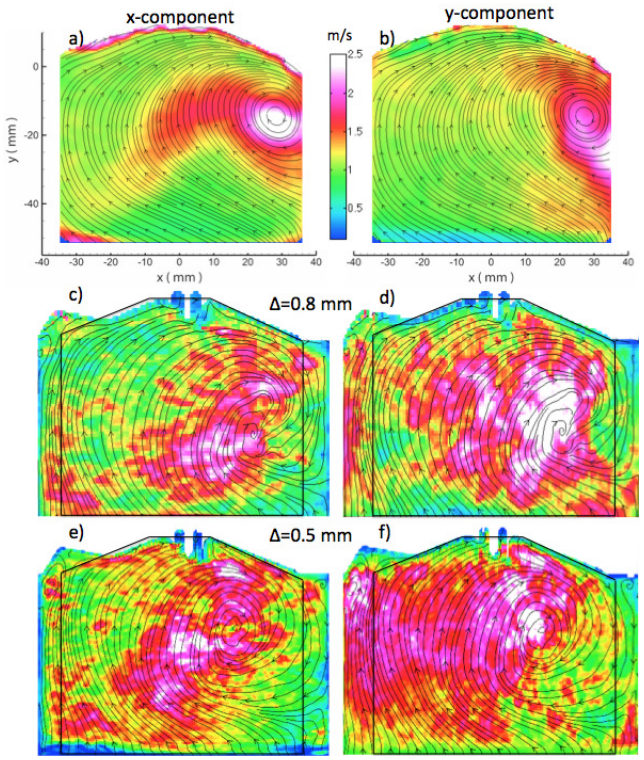

FIGURE 6.11: The standard deviation for the two measured velocity components at -90° CA (a,b). The corresponding RMS of the coarse simulation (c,d) ($\Delta = 0.8$ mm). The corresponding RMS of the medium simulation (e,f) ($\Delta = 0.5$ mm).

have been presented, showing that the grids are fine enough for the simulations.

Comment: The determination of LES quality based on the ratio of turbulent to laminar viscosity is not adequate since it depends highly on the chosen numerical schemes. In Chapter 7, based on the large numerical results of engine simulation using different numerical schemes, we show that the common 'LES quality criteria' are insufficient to determine 'good LES'.

6.6.2 Fired-case at 800 rpm

In the fired-case, the predicted pressure curves match the experimental data rather well as it can be seen in Fig. 6.16b (private communication with Brian Peterson from Darmstadt University). During the expansion stroke, some deviations between the simulated and the experimental pressure curves can be seen in Fig. 6.16b. This discrepancy may be due to the unknown wall temperature, according to our knowledge it is quite sensitive for the combustion case [210, 211]. A further reason may be the neglect of the effect of the unburned gas temperature on the laminar flame speed in the present study, this

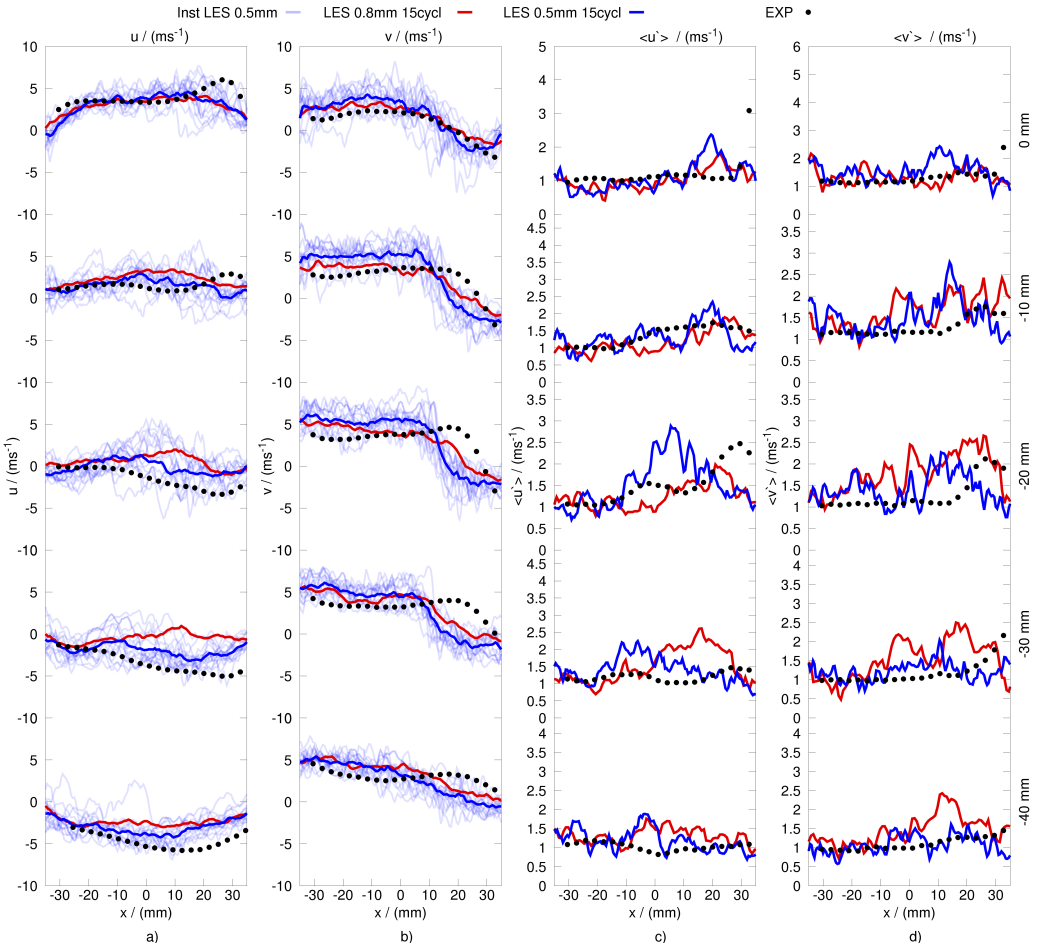

FIGURE 6.12: Predicted and measured velocity profiles at -90° CA. Lines in sub-figures (a) and (b) represent the phase averaged horizontal velocity component u and the vertical velocity component v. Additionally, 15 instantaneous velocity profiles obtained by the simulation on the medium grid are shown. Sub-figures (c) and (d) illustrate the corresponding RMS of u and v, respectively. Experimental results are shown with symbols. Results are shown for the horizontal sampling lines at 0 mm, -10 mm, -20 mm, -30 mm and -40 mm (see Fig. 6.5).

may affect the turbulent flame speed as well as the burning duration and lead to the deviation in the pressure curves in Fig. 6.16b.

A comparison of the simulated and measured flame propagation within a horizontal swirl plane (Fig. 6.17) is shown in Fig. 6.18 for different crank angle degrees (-6° CA to 4° CA). The left side of Fig. 6.18 shows the flame propagation as obtained in the experiments for -6° CA to 4° CA (private communication with Brian Peterson from Darmstadt University). The right side of the figure shows the images of the reaction progress variable source term from the simulation on the medium grid with a cell size of $\Delta = 0.5\,\text{mm}$. It is found that the general trend of the flame propagation is well-captured, the wrinkles of the flame front in the measurement and the simulation are very similar

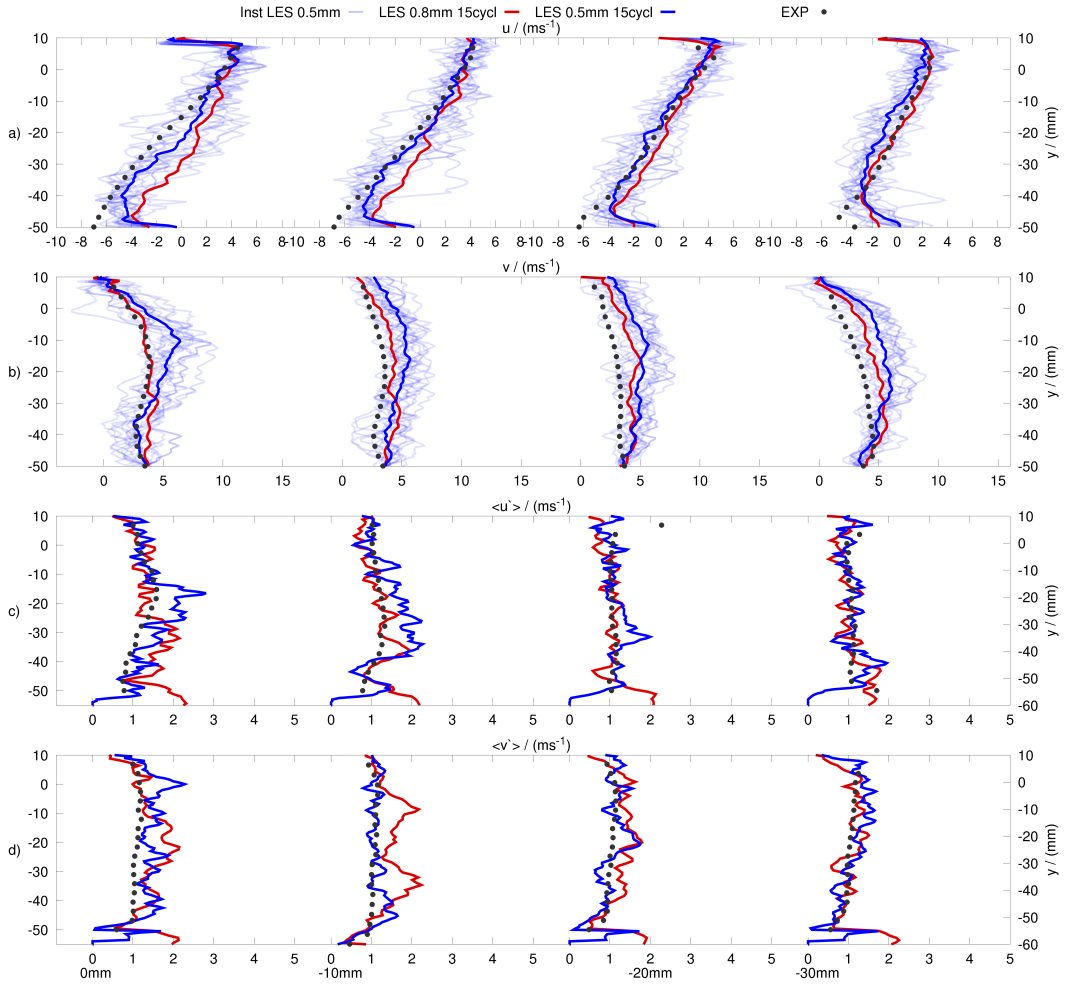

FIGURE 6.13: Predicted and measured velocity profiles at -90° CA. Lines in sub-figures (a) and (b) represent the phase averaged horizontal velocity component u and the vertical velocity component v. Additionally, 15 instantaneous velocity profiles obtained by the simulation on the medium grid are shown. Sub-figures (c) and (d) illustrate the corresponding RMS of u and v, respectively. Experimental results are shown with symbols. Results are shown for the vertical sampling lines at 0 mm, -10 mm, -20 mm and -30 mm (see Fig. 6.5).

in structure and scale. This also indicates that the turbulence of the in-cylinder flow is likely to be well-resolved by our approach. The applied FSD model [32] shows a reasonable agreement between the numerical simulation and the measurement for the in-cylinder pressure and the flame propagation.

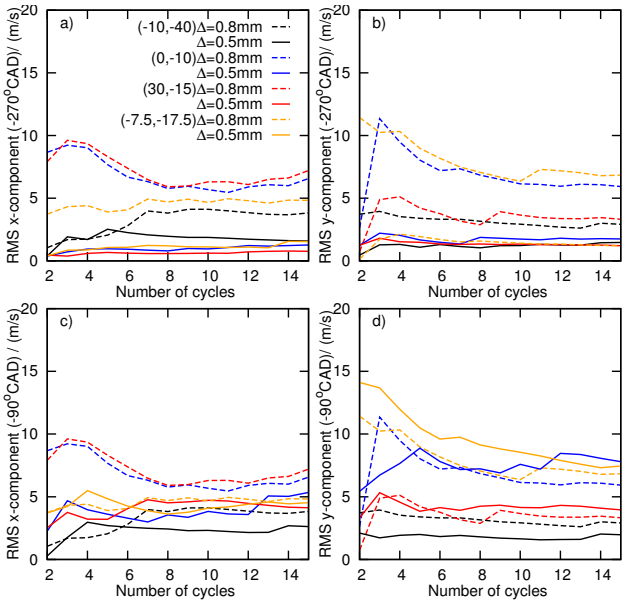

FIGURE 6.14: The standard deviation in the tumble plane for two grids ($\Delta = 0.8$ mm and $\Delta = 0.5$ mm) at -270° CA (a,b) and -90° CA(c,d). The coordinates of these points are p1(-10,-40), p2(0,-10), p3(30,-15), and p4(-7.5,-17.5) mm.

6.6.3 Numerical results for the motored-case at 1500 RPM

To demonstrate the suitability of the proposed approach for HPC, a fine grid engine simulation was performed on a domain containing 180 million cells with a size of 0.2 mm. Employing 8196 CPUs on the supercomputer JUQUEEN, the computation required around 3.15 million cpu-h to complete a single cycle. Since the simulation is computationally very expensive, only one cycle was computed. We show the coarse and medium cycles' averages combined with the finer results. The fine simulations are not meant to give fully converged statistics but rather additional evidence that a refinement will not change the results much, i.e. that results are largely converged, and give an indication of information that might not be available on a coarse grid. Figures 6.19 and 6.22 show the comparison between the simulated instantaneous velocity profiles, 150 measured instantaneous and the corresponding measured mean velocity profiles at -270° CA and -90° CA. A good principal agreement between the measured and simulated velocity structures can be observed as the simulated velocity profiles are well aligned inside the measured velocity profile envelope, the small structures of the flow field are clearly visible. The comparison between the simulated and the measured instantaneous velocities at -270° CA and -90° CA is shown in Figs. 6.20, 6.21, 6.23, and 6.24. It is interesting to observe that the level of deviation from the ensemble average and the length scales are

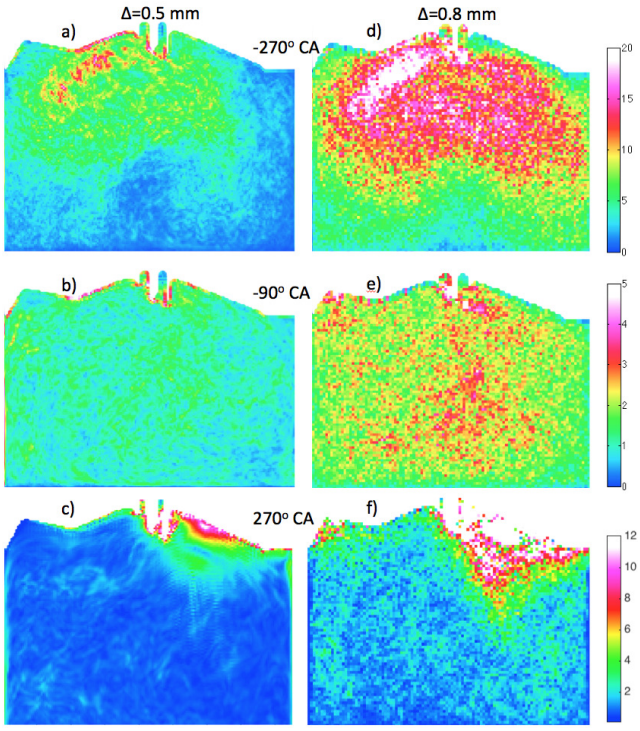

FIGURE 6.15: The ratio of turbulent to laminar viscosity (ν_t/ν) in the tumble plane for the 0.5 mm (a,b,c) and 0.8 mm (d,e,f) grid at -270° CA, -90° CA, and 270° CA.

very consistent (Figs. 6.23c, 6.23d, 6.24c, and 6.24d). One can also see the high resolution that is achieved by the LES on the 0.2 mm grid compared to the PIV measurements with a resolution of 1.8 mm as given by the experimentalists (Figs. 6.23 and 6.24).

6.7 Discussion

6.7.1 Numerical efficiency

The presented technique is simple and efficient for engine LES using a particle description of the moving walls. Depending on the resolution of the computational domain, the number of particles that is required for the description of the geometry is different, as shown in Table 6.5. For a system containing N_p particles, $\Theta(N_p)$ operations are required for the numerical treatment, including solving an ODE for each particle, blocking the particle-containing cells and assigning the velocity and accelerations for the particles. A smaller number of operations, which is usually one order of magnitude lower than $\Theta(N_p)$, is applied to impose the flow conditions to the solid cells at the boundaries.

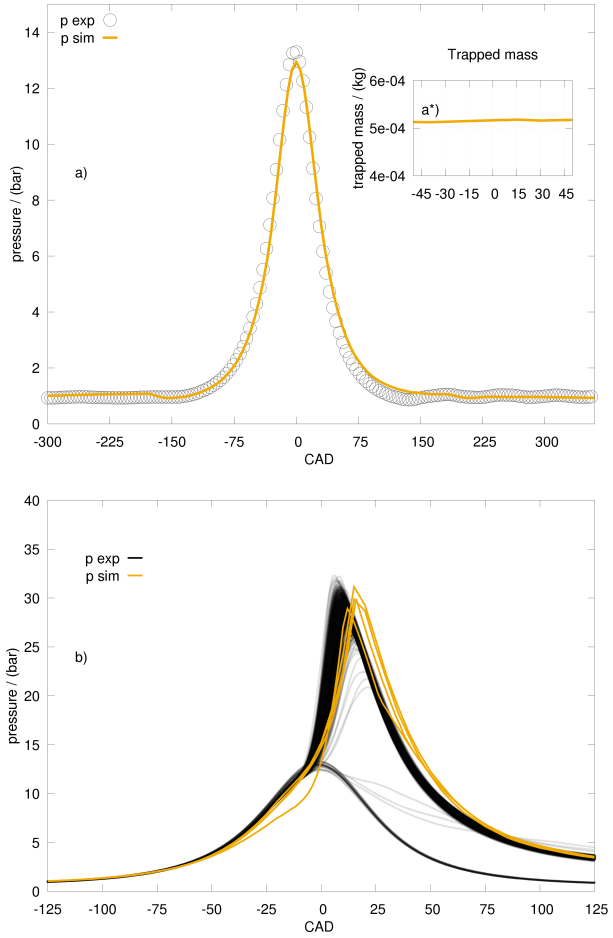

FIGURE 6.16: Comparison of the in-cylinder pressure obtained from LES simulation without (a) and with (b) combustion against measurements. The mass conservation is illustrated in (a*) by plotting the trapped mass during compression and expansion.

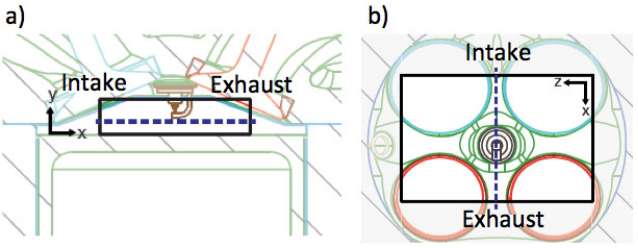

FIGURE 6.17: The vertical position of the interrogation window (a) and the size of the interrogation region (b) (private communication with Brian Peterson from Darmstadt University).

In fact, the number of operations related to the Lagrangian particles is time-dependent, and often not more than 2/3 of the particles are used to handle the moving boundaries. For example, during the intake stroke (after the valve overlap period), the particles describing the exhaust valves are not used. During the compression stroke, the numerical

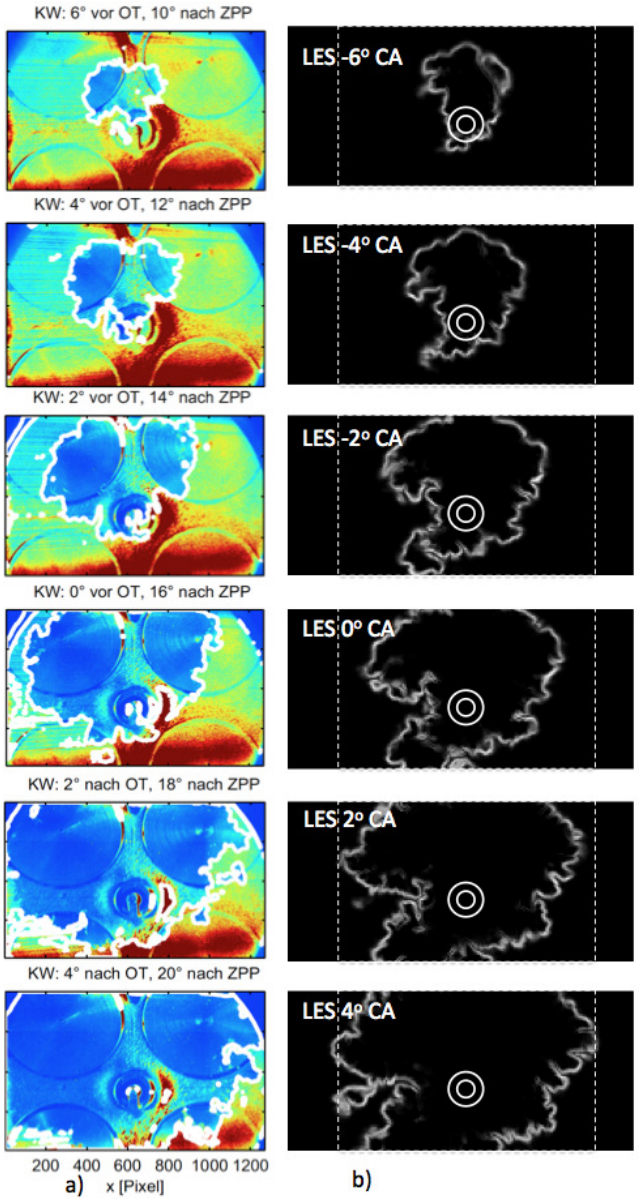

FIGURE 6.18: Images of the flame propagation within a horizontal swirl plane (Fig. 6.17) from (a) the experiment (view from the top of the engine, the two intake ports are shown in the upper part of the image, and the two exhaust ports are in the lower part) (private communication with Brian Peterson from Darmstadt University) and (b) the simulation ($\Delta = 0.5$ mm).

treatments are applied only for the particles describing the piston while both the particles describing the intake and the exhaust valves are not taken part in any numerical operations. Figure 6.25 shows the number of particles which are used in a complete cycle simulation of a four-stroke engine.

Figure 6.25 illustrates (for three grid sizes) the ratio of the total number of particles to the number of Eulerian cells (Fig. 6.25a), the ratio of the number of particles describing the boundaries to the total number of particles (Fig. 6.25b), and the ratio of used particles

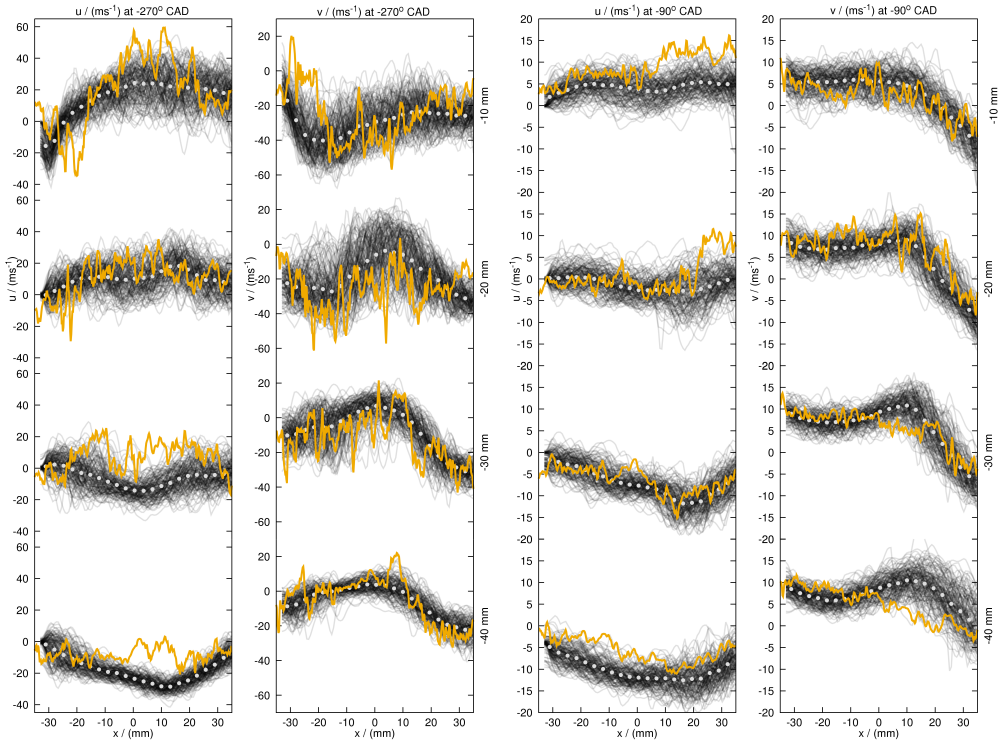

FIGURE 6.19: Velocity profiles obtained on the fine grid; measured instantaneous (—), measured mean (○) and simulated (—) on the fine grid. The results are shown for the horizontal sampling lines at -10 mm, -20 mm, -30 mm and -40 mm (see Fig. 6.5).

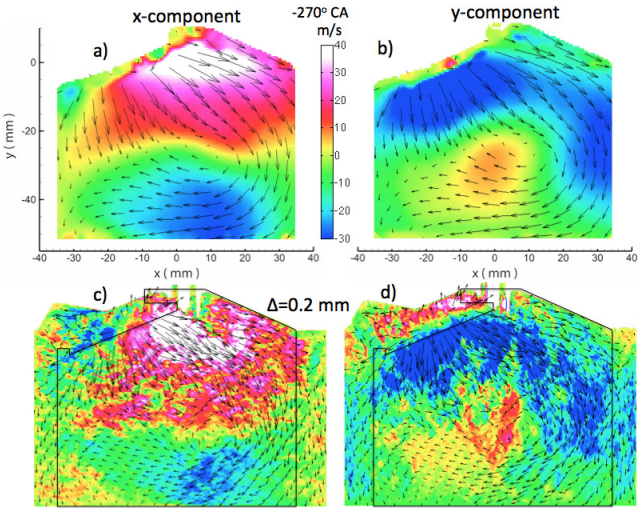

FIGURE 6.20: The phase-averaged measured (a,b) and the instantaneous simulated (c,d) velocity components obtained on the fine grid at -270°CA.

to the total number of particles in one cycle of the engine simulation (Fig. 6.25c). Compared to the total number of numerical operations that are needed for solving the Navier-Stokes equations $\Theta(N_{\text{cell}})$, where N_{cell} is the number cells of the computational domain, the number of calculations $\Theta(N_p)$ required by our proposed method is small: $\Theta(N_p) \subset \Theta(N_{\text{cell}})$. Additionally, this approach avoids the numerical overhead, and

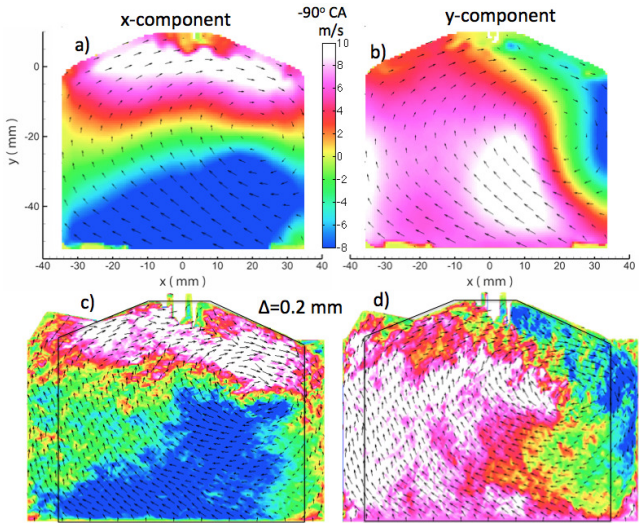

FIGURE 6.21: The phase-averaged measured (a,b) and the instantaneous simulated (c,d) velocity components obtained on the fine grid at -90°CA.

Test cases	Number of particles
$\Delta = 0.8$ mm	73864
$\Delta = 0.5$ mm	246840
$\Delta = 0.2$ mm	2988360

TABLE 6.5: Number of particles (N_p) in different grid sizes.

the decomposition of the computational domain is simple and independent from the geometry, while it is efficient in massive parallel simulations (demonstrated using up to 8196 cores). The distributions of the ratio between the number of particles and the number of cells per CPU in one time step in the three test cases are demonstrated in Fig. 6.26. Obviously, the particles are unequally distributed over the CPUs since the moving boundaries are located in certain areas of the computational domain.

6.7.2 Limitations

Although the proposed approach is simple, flexible, robust, and computational efficient, there are certain limitations that are necessary to be addressed in the scope of this paper.

Firstly, the wall treatment is less than second order accurate (in terms of wall location), which must however be put in perspective to an overall very high grid resolution and

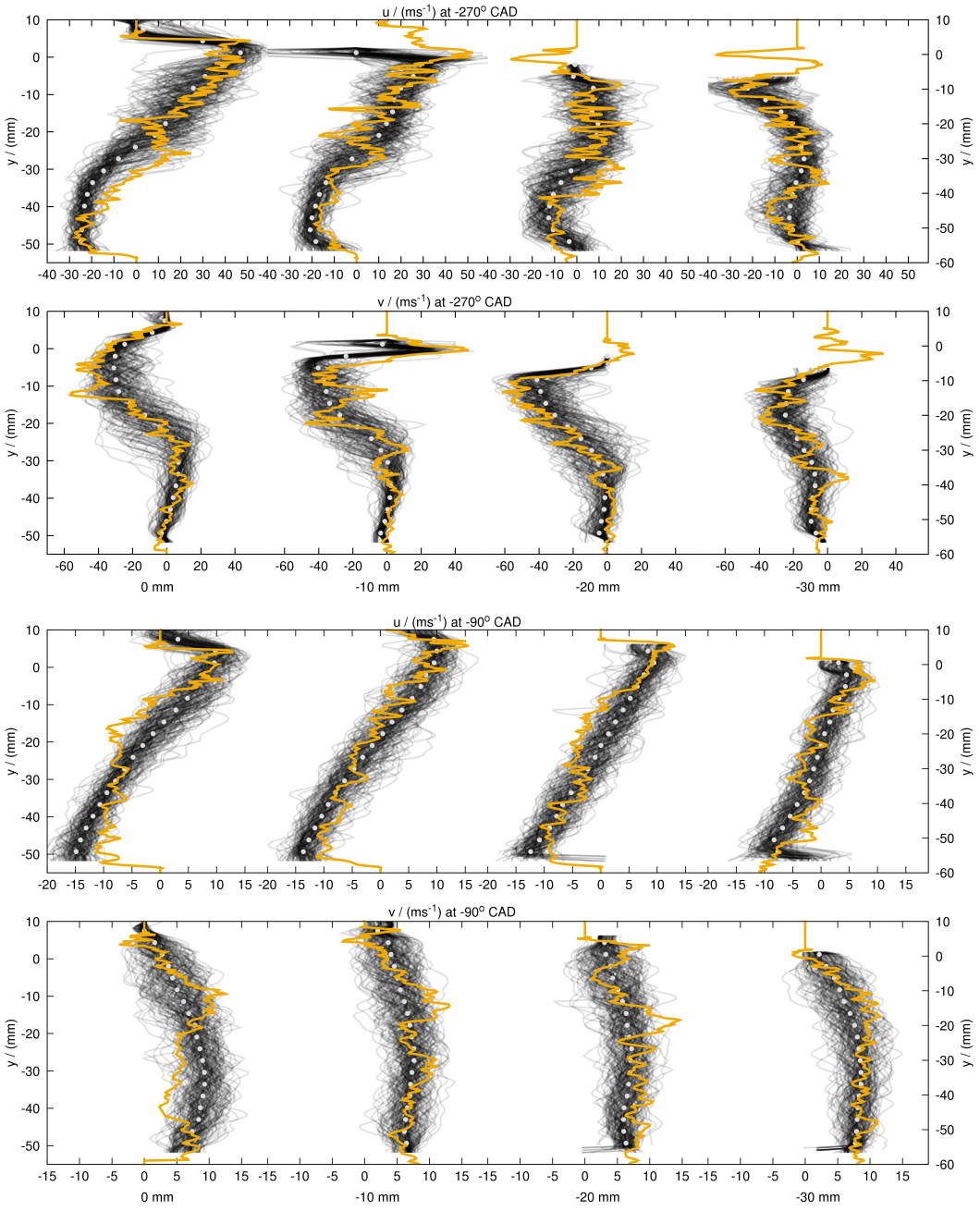

FIGURE 6.22: Velocity profiles obtained on the fine grid; measured instantaneous (—), measured mean (○) and simulated (—) on the fine grid. The results are shown for the vertical sampling lines at 0 mm, -10 mm, -20 mm and -30 mm (see Fig. 6.5).

the uncertainties involved in the wall modelling of a strong recirculating, highly unsteady flow. Overall, wall treatment is still very much a field of research in IC engine simulations.

Secondly, the major drawback of the proposed method is the limit in resolving the near wall region at fixed and moving boundaries, which may influence the flow and the combustion away from the walls. Obviously, without local refinement, having proper

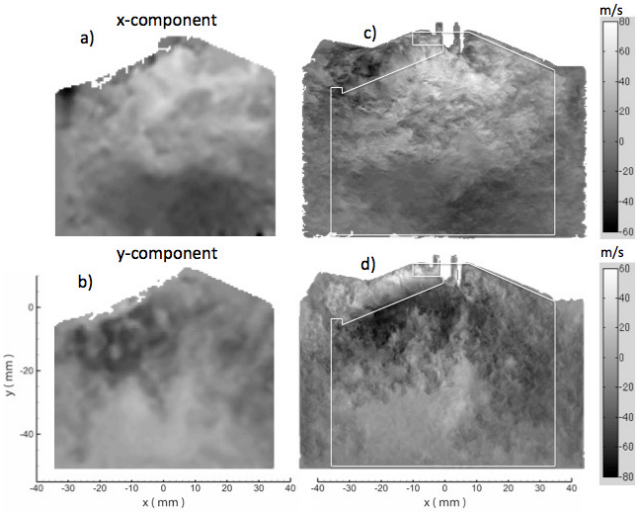

FIGURE 6.23: The instantaneous measured horizontal (a) and vertical (b) velocity components at -270°CA. The corresponding simulated instantaneous velocity components (c,d) were obtained on the fine grid.

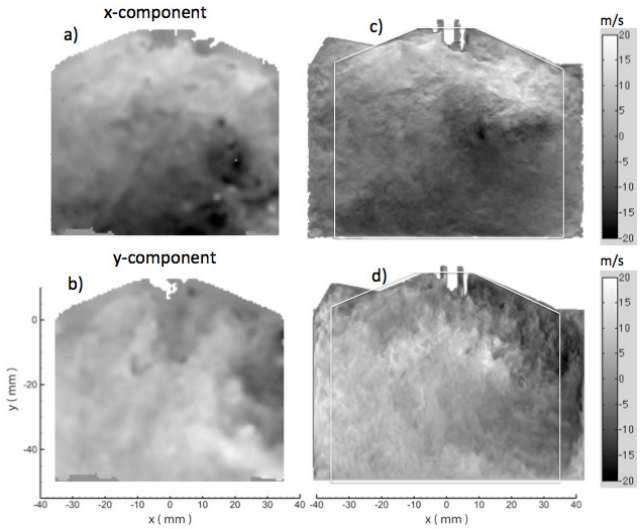

FIGURE 6.24: The instantaneous measured horizontal (a) and vertical (b) velocity components at -90°CA. The corresponding simulated instantaneous velocity components (c,d) were obtained on the fine grid.

geometric description of the moving surfaces would require more cells, a bigger computational domain and more CPUs. This also leads to a situation when the valve gaps are smaller than the cell/filter size (Δ) and therefore are not resolved. In engine simulation, the description of the "valve lift profile" is crucial to capture the in-cylinder flow processes, as shown by Cleary et al. [212], Davis et al. [213] and Kreuter et al. [214]. Typically, the small valve gap is resolved by a local refinement in the region near the valve seats. However, the importance of resolving the valve gaps down to small

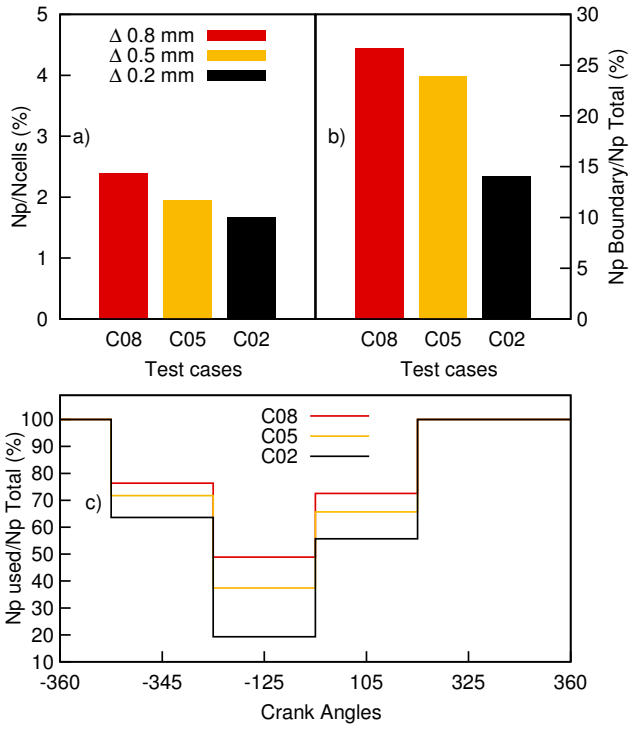

FIGURE 6.25: Ratio of the total number of particles (N_{pt}) to the total number of computational cells (N_c) (a). Ratio of the number of boundary particles (N_{pb}) to the total number of particles (b). Percentage of the used particles (N_{pu}) over a complete engine cycle simulation (c).

value in the range of 0.1-0.125 mm at the beginning of the intake stroke is still questionable. As the intake flow velocity and the piston speed are small, the turbulence that is generated during this short period (10-15 CADs) might not have a strong influence on the overall turbulence of the in-cylinder flow later. It is also questionable whether the relevant turbulence is formed in the valve (seat) boundary layer, which would require a tremendously fine resolution, or rather in the free shear layers downstream of the valve, which requires a good resolution there, while reducing the need for very fine near wall resolution. Nevertheless, further studies on the effect of the small valve gaps on the in-cylinder flow field are necessary to gain better understanding on this issue.

6.7.3 Uniform grid vs. local refinement

The importance of local refinement is undeniable and has been pointed out in the previous sections. However, the uniform Cartesian grid has its own advantages in the LES applications. Classically, one would wish to have the finer resolution near valves while, to compromise with the limited computational resources, the number of cells in the

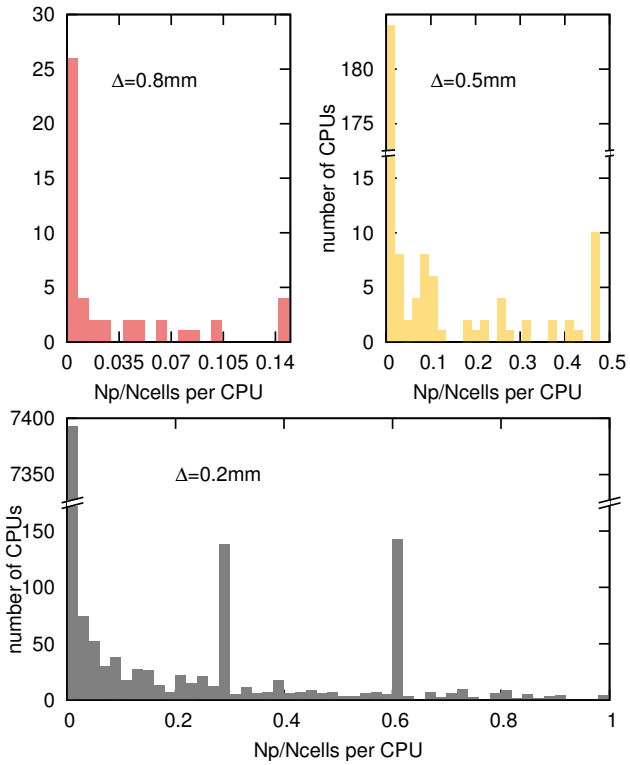

FIGURE 6.26: The distribution of the ratio of the number of particles (N_p) to the number of cells (N_{cell}) per CPU in the three test cases.

combustion chamber domain must be reduced. Achieving this through increasing the cell size elsewhere leads to artificial dissipation. However with LES, a fine resolution in the combustion chamber and hardly dissipative schemes (e.g. 2nd order central on Cartesian grids) are of paramount importance to avoid artificial dissipation and a too low turbulent flame speeds there. Our paper shows one compromise and what this compromise will result in. We would however like to point out that the compromise of very fine resolution near the valves and a coarse grid in the cylinder is likely to fail with LES, while using very fine grids everywhere is not feasible. A further reason for the artificial or numerical dissipation is a deformation of the cells, such as stretching, compressing or distorting. Moreover, using different cell sizes for the same domain in the LES context causes a commutation error of $\mathcal{O}(\Delta^2)$ with the filter width Δ (Ghosal et al. [215]). These errors cannot be afforded inside the cylinder, where numerical diffusion would lead to insufficient flame wrinkling, under predicted turbulent flame speed and hence wrong pressure curves and burn out data. Therefore, concentrating the computational effort near the valves and sacrificing the accuracy inside the cylinder may not be a good approach for the LES of an IC engine. Since the in-cylinder flow is driven by inertial forces, pressure difference, and piston motion in a complex geometry, having an equal

distribution of computational cells everywhere in the combustion chamber appears to be an appropriate approach to capture the flow dynamics correctly, to resolve a wide range of turbulent scales and to avoid the numerical dissipation. Based on the results obtained on the coarse, medium, and fine meshes, our approach demonstrates the capability of capturing the flow field correctly, even though the smallest valve gap is restricted by the cell size. Our approach, while being very promising with capable computer resources, is computationally costly overall, since a certain "minimum" number of cells is needed to resolve the geometry - so cheap test runs on very coarse grids are not feasible with it. It is however clear that the current progress in computational power is mainly driven by further vectorisation and parallelization, for which our approach is well suited. Table 6.6 presents the computational costs from different test cases (coarse/medium grids, slow/fast motor speed) of the Darmstadt engine [141]. It becomes clear that the computational costs are reduced significantly when the motor is operated at high RPM. This is a direct result of using a compressible scheme with a small CFL number to resolve acoustic wave propagation - which is often not resolved by less costly simulations.

Test cases	CPU hours per cycle
$\Delta = 0.8$ mm, 800 RPM	4608 cpuh/cycle
$\Delta = 0.8$ mm, 1500 RPM	2250 cpuh/cycle
$\Delta = 0.5$ mm, 800 RPM	40320 cpuh/cycle
$\Delta = 0.5$ mm, 1500 RPM	20160 cpuh/cycle

TABLE 6.6: Computational cost for different grid sizes and motor speeds.

6.8 Conclusions

We have presented the application of LES for simulations of a four-stroke engine with compressible flow, combustion modelling and moving boundaries. The numerical results show a good agreement with the experimental measurements in the case of pure gas exchange, as well as under fired conditions. During pure gas exchange, the flow is driven mainly by the inertia of the fluid flow and the piston and valve motion. The pressure and the velocity profiles in this operating state show a good agreement with the measurements for all applied grid spacings, which confirms the accuracy and efficiency of the implemented numerical methods. Furthermore, as shown by the numerical results,

different grid sizes lead to a very similar amount of the predicted turbulent kinetic energy. This give evidence for the robustness of the applied method.

The simulation of the fired engine showed a well-captured flame propagation. The flame front geometry obtained in the simulation shows a clear similarity to the experiment, similarity in terms of the volume of the flame kernel, its level of corrugation and its propagation speed. The predicted pressure matches the measured pressure of the fired-case. This indicates further the adequateness of the implemented models and the numerical solution, as the flame propagation depends strongly on the turbulence.

The operations describing the particle motion are simple to implement, low in the required number of operations and easy to parallelize - in particular on an isotropic and (i, j, k)- countable computational grids, where only a few hashing operations are required in order to calculate the logical particle location. The strategy of using Lagrangian particles for describing the moving parts inside the engine was proved to be accurate and of high algorithmic efficiency.

Chapter 7

Investigation of Numerical Effects on the Flow and Combustion in LES of ICE

Authors: T. Nguyen, A. Kempf

This chapter including all figures and tables was previously published in Oil & Gas Science and Technology - Revue d'IFP Energies nouvelles Volume 72, Number 4, July - August 2017, Investigation of Numerical Effects on the Flow and Combustion in LES of ICE, Thuong Nguyen and Andreas Kempf. The author T. Nguyen developed the code, ran all the simulations, wrote the paper and generated all figures and tables. The author A. Kempf contributed corrections, discussions and proof-reading.

7.1 Abstract

This work investigates the influence of numerical dissipation on the modeled combustion in Large Eddy Simulations (LES). It is well known that capturing the dynamics of the in-cylinder flow is crucial for engine simulations, as it strongly affects flame propagation. The flame propagation during the power-stroke highly depends on the turbulence level that is developed throughout compression. This turbulence level will be strongly influenced by the accuracy of the numerical schemes employed. Even a small extent

of upwinding, filtering, low-order implicit time-stepping, cell-stretching or mapping between grids may affect the flow field, the turbulence level, and hence the turbulent flame speed and the pressure curve. To provide a reference, the LES in-house code PsiPhi is used, which ensures a minimum of dissipation due to high order explicit time-stepping, homogeneous and isotropic filters and cells. Good stability of the code permits the use of a second-order central differencing scheme (CDS) for the transport of momentum, avoiding numerical dissipation. To analyse the effect of numerical dissipation, simulations of a fired engine are performed using different numerical schemes for the convection of momentum. Physical quantities including the total kinetic energy, the velocity gradient, the turbulent viscosity, the in-cylinder pressure, the flame propagation or the burning rate of different test cases are evaluated and compared to each other to show the numerical effects on combustion. Furthermore, the suitability of common LES quality criteria including an energy criterion and viscosity ratio is discussed based on the comparison of simulations with less and more accurate numerics. It is shown that these LES quality indicators can be highly misleading.

7.2 Introduction

The extensive use of CFD in numerous engineering applications leads to many fundamental questions regarding the applied numerical approaches themselves, especially the effects of the numerical schemes on the simulation results, which have been investigated in many studies, as for examples [216, 217, 218, 219]. Obviously, reducing the numerical errors is essential for an accurate solution of the transport equations of different physical quantities [220, 221]. It is clear that to obtain more accurate numerical results, fine grids and high order numerical schemes are preferable. However, high simulation costs due to finer computational grids as well as the numerical instability resulting from high order schemes are factors that must be considered. As a matter of fact, a compromise between accuracy, stability and convergence is often necessary. Depending on the nature of the problem, one may try and choose suitable numerical schemes that provide good results.

In order to investigate the influence of numerical schemes applied to engine simulations, several nominally 2nd-order numerical schemes are employed to examine their effect on the simulated in-cylinder flow field, as well as the combustion process. In Large-eddy simulation, investigators often outline the importance of the filter size and high-order

discretization schemes to yield reasonable simulation results. However, it should be stressed that the order of numerical schemes is just one part of the story: It has been observed in many studies [222, 125, 223, 221] that two schemes with the same nominal order of accuracy can yield very different results.

In the context of engine simulations, Misdariis et al. [224] investigated the influence of several numerical schemes for the discretization of the convective flux on the flow field and the combustion inside the engine. They observed a rather small difference between the in-cylinder flow fields resulting from the numerical schemes of second and third order of accuracy. In comparison to their findings, we observe a stronger impact of the applied numerical schemes on the small scale features of the simulation results, as illustrated in Fig. 7.1. The CDS, QUICK and the TVD schemes yield significant differences in the velocity field where small structures are found in the whole flow field for the simulation using CDS with 2^{nd} and 4^{th} order, but not in the simulation with the TVD schemes. Interestingly, in spite of distinctive flow structures between different simulations, a rather small discrepancy of the total kinetic energy of the in-cylinder flow between the simulation results of all the employed schemes is achieved. This may not be overly surprising, since most of the turbulent kinetic energy exists on the larger scales. However, significant differences are observed on the smaller scales and hence the modeled sub-grid energy that has much impact on the wrinkling factor and turbulent flame speed.

Comment: Strictly speaking, the CDS in this study is not a complete 4^{th} order since the interpolation of the fluxes at the cell faces is performed based on 4 data points. However, the integration of the fluxes at the cell face is a 2^{nd} order.

In the scope of this paper, the commonly used LES quality indices are also discussed. Interestingly, more dissipative schemes (like TVD) tend to cause a smaller turbulent viscosity due to smaller velocity gradients, as the numerical scheme itself is more dissipative. One may argue that with a TVD scheme, much of the dissipation is achieved by the discretization, so that the turbulence model needs to dissipate less energy. This leads to a misleading situation where many commonly used LES quality criteria must fail. Criteria based on the amount of modeled turbulent kinetic energy [74, 225], as well as the criteria based on the ratio between the turbulent and the laminar viscosity (as proposed by Celik et al. [209]), will no longer be adequate.

In this work, simulations of a four-stroke engine [141] with two intake and two exhaust valves are performed on two different computational grids with cell sizes of 0.8 mm and 0.5mm. To avoid the complexity of the mesh generation in a moving engine-geometry, the motion of the piston and the valves are described by Lagrangian particles in combination with an immersed boundary method (IBM) [164]. This mesh-free technique has been implemented in our code PsiPhi [156] and is simple and well suited for high-performance computing (HPC) applications. Flame propagation is modeled using a flame surface density model (FSD) [32] with iso-octane as surrogate fuel. The flow conditions at the intake and the exhaust ports of the engine are computed according to Navier-Stokes characteristic boundary conditions [136]. Our implementation [208] of Nicoud's Sigma model [87] is used to calculate the sub-grid scale stresses.

7.3 Numerical methodology for engine simulation

The Favre-filtered compressible Navier-Stokes equations for mass, momentum and total energy are discretized and solved numerically on an equidistant Cartesian grid using the finite-volume method. To handle the moving parts of the engine such as the piston, the intake and exhaust valves, an approach by Nguyen et al. [156] using Lagrangian particles and an immersed boundary method (IBM) [164] is applied for the description of the geometries as well as the interaction between the moving boundaries and the fluid. In this approach, the moving geometries are presented with a cloud of logic and massless particles instead of the unstructured cells from body-fitted methods. The harmonic motion of the piston and the motion of the valves from the valve lift profiles are governed by the transportation of different groups of Lagrangian particles, which are used to describe the moving objects. Extensive discussion about the advantages, limitations and implementation of the method can be found in [156].

7.4 Combustion modeling with Flame Surface Density

The modeling of the combustion process is described by the following equations:

$$\begin{aligned}\frac{\partial\overline{\rho}\widetilde{c}}{\partial t}+\frac{\partial\overline{\rho}\widetilde{c}\widetilde{u}_i}{\partial x_i}+\frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu_i}-\widetilde{c}\,\widetilde{u}_i)]\\ =\overline{\frac{\partial}{\partial x_i}\left(\rho D\frac{\partial c}{\partial x_i}\right)}+\overline{\dot{w}}_c=\overline{\rho S_d}\Sigma_{\text{gen}}\end{aligned} \tag{7.1}$$

$$\frac{\partial}{\partial x_i}[\overline{\rho}(\widetilde{cu_i}-\widetilde{c}\widetilde{u_i})]=\frac{\partial}{\partial x_i}\left(\overline{\rho}\frac{\nu_t}{Sc}\frac{\partial\widetilde{c}}{\partial x_i}\right)+\mathrm{F}_{\text{cgt}} \tag{7.2}$$

$$\begin{aligned}\overline{\rho S_d}\Sigma_{\text{gen}}\approx\rho_u S_l\Sigma_{\text{gen}}=\rho_u S_l\Xi|\nabla\overline{c}|-\mathrm{F}_{\text{cgt}}\\ \approx\rho_u S_l\Xi|\nabla\widetilde{c}|\end{aligned} \tag{7.3}$$

Comment: the counter gradient term F_{cgt} *is presented in section 2.7 of Chapter 2.*

On the LHS of Eq. 7.1, the sub-grid scalar flux $\partial/\partial x_i[\overline{\rho}(\widetilde{cu_i}-\widetilde{c}\,\widetilde{u}_i)]$ is calculated by the classical gradient hypothesis and a counter gradient term F_{cgt} (Eq. 7.2). In Eq. 7.3, the generalized flame surface Σ_{gen} is approximated by replacing the Reynolds-averaged progress variable $|\nabla\overline{c}|$ by the Favre-filtered one $|\nabla\widetilde{c}|$, where the counter gradient transport F_{cgt} is implicitly included (see the discussion by Ma et al. [103]). The two terms $\overline{\dot{w}}_c$ and D in Eq. 7.1 are the mean reaction source term and the molecular diffusivity. In Eqs. 7.2-7.3, the turbulent viscosity, the turbulent Schmidt number, the laminar flame speed and the wrinkling factor are represented by ν_t, Sc, S_l and Ξ, respectively.

The influence of turbulence on the flame propagation is described by the wrinkling factor Ξ, which is modeled using an approach proposed by Muppala et al. [32].

$$\Xi=1+a\mathrm{Re}_t^{0.25}\left(\frac{u'}{S_l}\right)^b\left(\frac{p}{p_0}\right)^c \tag{7.4}$$

The turbulent Reynolds number Re_t is a function of the sub-grid scale velocity fluctuation u' [107], the cell size Δ and the laminar viscosity ν:

$$\mathrm{Re}_t=\frac{u'\Delta}{\nu} \tag{7.5}$$

In Eq. 7.4, the model parameters are set to $b = 0.2$, $c = 0.2$, and $a = 0.46$ for the iso-octane/air flame [32, 106]. In the scope of this paper, elaborated ignition modeling is not considered. Instead, we set the progress variable c to 1.0 (burned area) in a small region below the spark-plug so that the flame can start developing and propagating. It is clear that a proper ignition modeling can be done in more sophisticated ways. However, this would add additional uncertainty to the present study and be beyond the scope of the present study. We have therefore chosen to just show the effect of the numerical scheme. It is also clear that some shortcomings of numerical schemes can be partially compensated by suitable models or model constants. Further information on combustion modeling can also be found in our previous paper [156].

7.5 Experiment

Data	Value
Engine speed	800 rpm
Compression ratio	8.5
Displacement volume	499 cm^3
Crevice volume	2 cm^3
Volume at top dead center (TDC)	66.5 cm^3
Bore	86 mm
Stroke	86 mm
Cylinder clearance height	2.6 mm
Intake valve closure (IVC)	-125 CA
Exhaust valve open (EVO)	105 CA
Exhaust valve close (EVC)	-345 CA
Intake valve open (IVO)	325 CA
Spark timing (ST)	-16 CA
Fuel	Iso-Octane
Equivalence ratio (ϕ)	0.833
Avg. pressure intake (p_{in})	0.95±0.02 bar
Intake temperature (T_{in})	47±3°C
Intake density (ρ_{in})	1.0344 kg/m^3

TABLE 7.1: Engine specifications in the fired case

In this study, we consider a four-stroke optical engine, which is operated by the Dreizler group at Technical University of Darmstadt [141]. The engine runs at 800 RPM with iso-octane (C_8H_{18}) as a surrogate fuel. Optical access is provided via a quartz glass cylinder liner and a flat piston window. The engine consists of a twin-cam, an overhead-valve pent-roof cylinder head with two intakes and two exhaust valves. The characteristics

and operation point are shown in Table 7.1. For further details of the engine, the readers are referred to the paper of Baum et al. [141].

7.6 Numerical Setup

The LES in-house code PsiPhi [162, 157, 156, 36, 187] is used to solve the Favre-filtered governing equations for a compressible fluid. An equidistant Cartesian grid is used throughout the whole computational domain, which provides good accuracy and avoids the numerical dissipation caused by deformed or stretched cells of an unstructured-grid. In an equidistant Cartesian grid, the numerical error directly results from the applied interpolation scheme without adding the truncation error from the irregular grid. The disadvantage of this approach is that a local refinement is not possible, which leads to an increase of the total number of cells in the computational domain.

Applied Numerical scheme	Limiter functions
CDS, 2^{nd} order	-
CDS, 4^{nd} order	-
QUICK	-
Limited Linear	$\Psi = \max\left[0, \min\left(2r/\beta, 1\right)\right], \quad \beta = 0.1$
TVD-CHARM	$\Psi = r(3r+1)/(r+1)^2$ if $r > 0$ and $\Psi = 0$ if $r \leq 0$
TVD-vanLeer	$\Psi = (r + \lvert r\rvert)/(1 + \lvert r\rvert)$

TABLE 7.2: The numerical schemes used in the engine simulations

Results from different numerical schemes are compared: a 2^{nd} and a 4^{th} order central differencing schemes (CDS2, CDS4), the limited linear scheme [125] (LL01), the QUICK scheme [118] and two total variation diminishing schemes using CHARM [120] (TVD-CH) and Van Leer [121] (TVD-VL) limiters. It is important to note that we use a value β=0.1 for the limited linear scheme, which makes this scheme less dissipative and supposedly 'close to CDS' (see Table 7.2).

The TVD scheme with CHARM limiter [120] is used for the discretization of all scalar quantities including density $\overline{\rho}$, progress-variable $\overline{\rho}\tilde{C}$ and total energy $\overline{\rho}\tilde{E}$.

Comment: in the motored-cases, the deviation of the local values of the scalar quantities is quite small. Therefore, the discretization of these quantities may not have strong effects on the numerical results. In the fired-cases, where the large deviation of the scalar quantities between the burned and unburned regions is present, significant impact

of the discretization of the scalars on the numerical results must be considered. Due to a large number of engine simulation in this study, we do not consider the effect of the discretization of the scalar quantities. However, further study on this issues is important and necessary.

An explicit third order Runge-Kutta scheme is employed for time integration. The unresolved turbulent viscosity is determined with Nicoud's σ model [87] with the model constant C_m of 1.5. In our experience, the combination of a sub-grid model, good equidistant Cartesian grids and an explicit low storage Runge-Kutta scheme with a small CFL number is sufficient to achieve stability - some upwinding is only introduced right at the walls and for increased Mach number. This approach has enabled stable - but not always easy to setup.

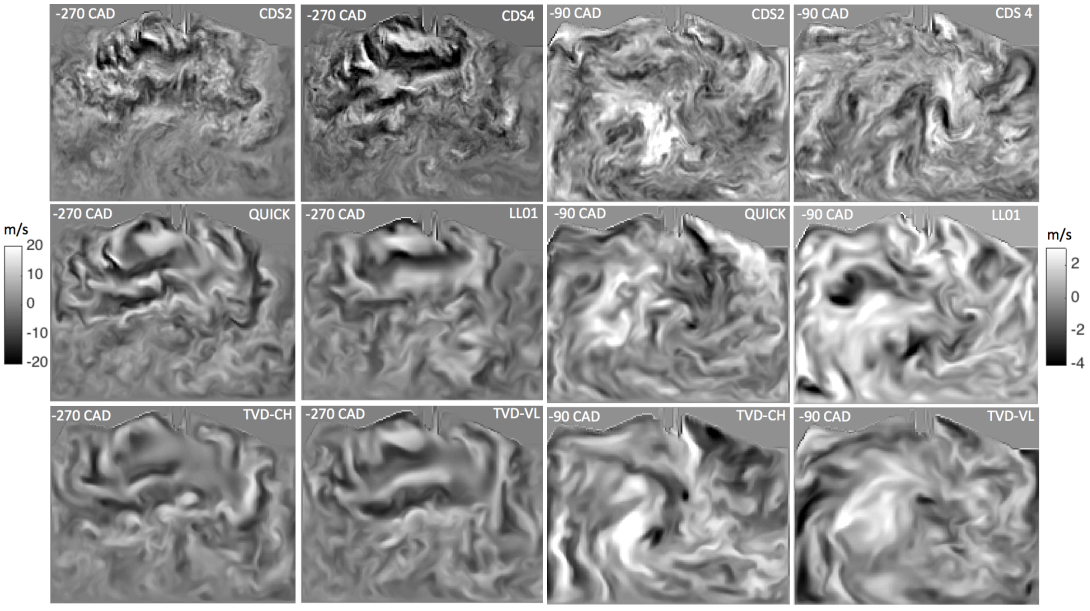

FIGURE 7.1: Comparison of the image-normal velocity component at -270 and -90 CAD for different numerical schemes from a chosen engine cycle

It is noted that the numerical analyses in the following sections are based on the phase-averaged data of five consecutive engine cycles for each numerical scheme. Nevertheless, we demonstrated some result from a chosen engine cycle to highlight the significant distinctions of the numerical results due to the different numerical schemes.

7.7 Kinetic energy analysis

Figure 7.1 shows the normal velocity components at -270 and -90 CAD, which result from six different numerical schemes. In comparison to the simulated results obtained by CDS2 and CDS4, where small structures are dominant in the flow field, much smoother flow fields are produced by the simulations using the TVD schemes (TVD-CH, TVD-VL, LL01). On the other hand, only big structures are visible for the simulations with the TVD schemes for the momentum at -90 CAD. It is noticeable that the flow structures obtained by QUICK are somewhat in between CDS and TVD. One should note that the use of non-central schemes for momentum transport in LES is a fairly recent phenomenon, which has emerged in recent years with the availability of 'LES models' in robust mainstream codes like ANSYS [226], STAR-CCM+ [227] or OpenFOAM [228].

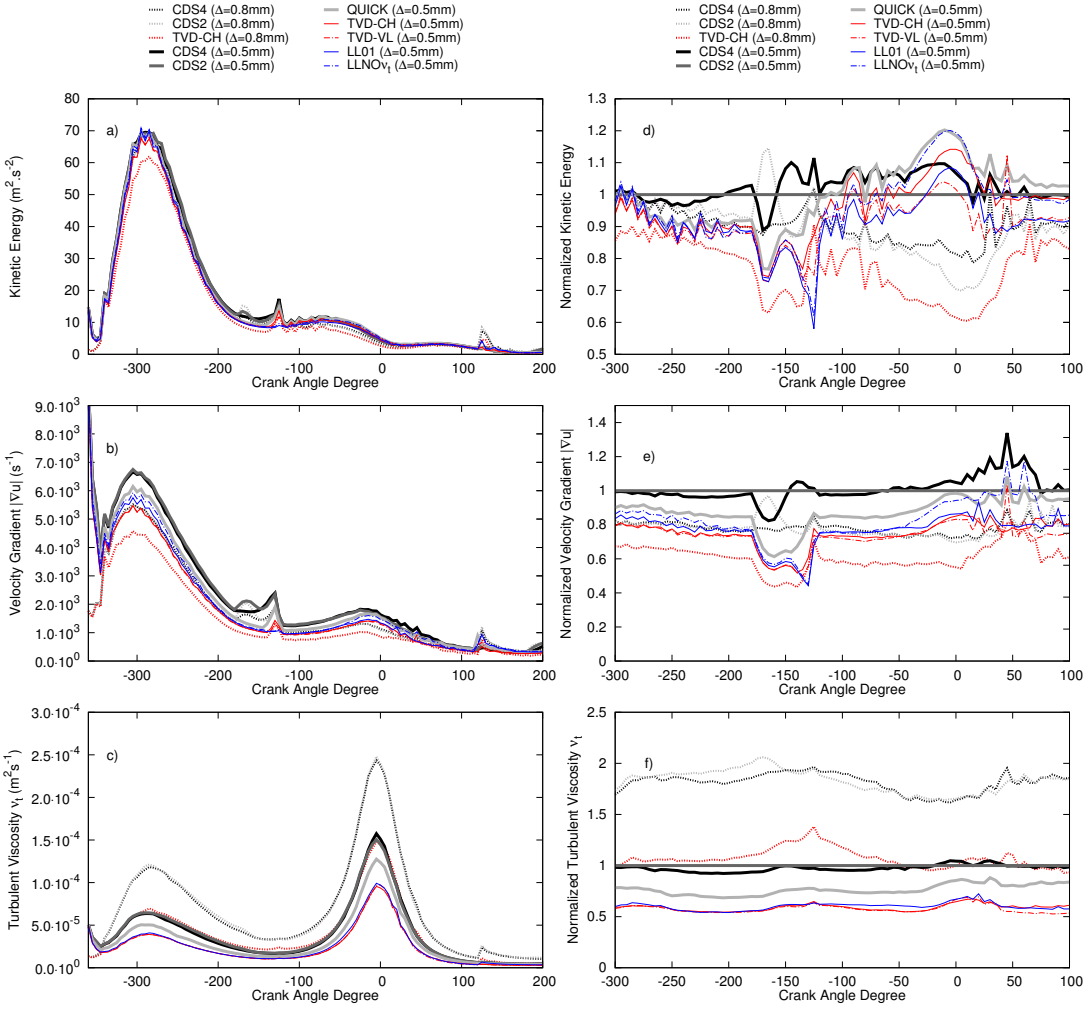

FIGURE 7.2: Comparison of the integrated kinetic energy KE_Ω, the integrated velocity gradient $|\nabla u|_\Omega$ and the integrated turbulent viscosity $\nu_{t\Omega}$ between different numerical schemes (absolute values on the left and relative values on the right)

As observed in Fig. 7.1, the simulations using CDS with 2^{nd} and 4^{th} order schemes resolve more small structures in the in-cylinder flow field than the other schemes. In order to quantitatively analyze the differences, we evaluate the integral of the kinetic energy, $KE_\Omega = (1/2)\int_\Omega (u_i^2) d\Omega$, the integral of the velocity gradient $|\nabla u|_\Omega = \int_\Omega [(\partial u_i)^2 + (\partial u_j)^2 + (\partial u_k)^2]^{0.5} d\Omega$ and the integral of turbulent viscosity $\nu_{t\Omega} = \int_\Omega \nu_t d\Omega$ inside the cylinder Ω. Figure 7.2 shows the comparison of the integrated kinetic energy KE_Ω, integrated velocity gradient $|\nabla u|_\Omega$ and integrated viscosity $\nu_{t\Omega}$ between six numerical schemes for momentum. As it is shown in Fig. 7.2a, the total resolved kinetic energy during engine operation is quite consistent between different numerical schemes, where the maximum value of KE_Ω is achieved around -270CAD and gradually decreases towards the compression and expansion strokes. The spikes of KE_Ω observed at -125 CAD and 105 CAD result from closing the intake valves and opening the exhaust valves, which cause a high velocity in the small gap between the valves and port walls. Interestingly, it is hard to notice any significant difference in the integrated kinetic energy KE_Ω of the in-cylinder flow field between the simulations using different numerical schemes. Despite the difference in the flow structures, all the numerical schemes seem to perform very well in resolving the kinetic energy of the flow field. This may not be surprising, since most of the kinetic energy exists on the largest scales.

Since the wide dynamic range of data values obtained during engine simulation makes it difficult to distinguish the differences of KE_Ω, $|\nabla u|_\Omega$ and $\nu_{t\Omega}$ resulting from different numerical schemes (Figs. 7.2a, 7.2b and 7.2c), using the normalization of all simulated data to the corresponding data obtained with the CDS2 scheme is a convenient way to highlight the variation between the schemes. As it can be seen in Figs. 7.2a and 7.2d, the resolved kinetic energy yielded by the six numerical schemes using 0.5 mm grid are not significantly different from each others, whereas on the coarse grid of 0.8 mm, the resolved kinetic energy is clearly lower. As seen in Fig. 7.2d, during the intake stroke, the total resolved kinetic energy of the simulations using more dissipative schemes such as QUICK, TVD-CH, TVD-VL and LL01 is lower than the resolved kinetic energy obtained by simulations using CDS2 and CDS4. However, during the compression stroke, KE_Ω obtained by simulations using QUICK, TVD and LL01 seems to dissipate quite slowly in comparison to KE_Ω obtained by CDS2 and CDS4. This may explain why we see an increasing trend of KE_Ω from QUICK, TVD and LL01 with reference to KE_Ω from simulations using CDS. Despite the significant differences in the flow fields obtained by

different numerical schemes (Fig. 7.1), the resolved kinetic energy KE_Ω of the in-cylinder flow shows no substantial difference for six studied numerical schemes. Obviously, the higher or lower value of resolved kinetic energy KE_Ω give no definite indication whether the applied numerical scheme is more or less accurate and dissipative than other schemes. Hence, it is questionable whether the resolved turbulent kinetic energy can be used to determine the quality of LES - or what LES quality actually is.

Comment: The resolved kinetic energy $KE = 1/2(U_i^2)$ *and the resolved turbulent kinetic energy* $k_{res} = 1/2(U_i - \langle U_i \rangle)^2$ *are two different but closely related quantities as they are directly calculated from the resolved flow velocity* U_i*. The root of the problem is that, the flow velocity strongly depends on the numerical schemes. Therefore, it is problematic for any assertion of LES quality, which relies solely on the obtained flow velocity. This study has shown that the determination of LES quality based on the TKE is highly questionable as the dissipative simulations always achieve the better LES quality.*

7.8 Velocity gradient and turbulent viscosity

We have shown that the resolved kinetic energy may not be adequate to differentiate the influence of numerical schemes on the in-cylinder flow field, even though clear differences between these schemes are observed in Fig. 7.1. On the other hand, the integrated velocity gradients $|\nabla u|_\Omega$ in Figs. 7.2b and 7.2f show a clear distinction between numerical schemes. As expected, the $|\nabla u|_\Omega$s obtained by both CDS2 and CDS4 are comparable and consistently larger than $|\nabla u|_\Omega$s of QUICK, TVD-CH, TVD-VL and LL01. Intuitively one may conclude that the turbulence level of the in-cylinder flow field obtained by QUICK (see Fig. 7.1) lies somewhere in between TVD and CDS schemes. This is also clearly shown in Figs. 7.2b and 7.2e where $|\nabla u|_{\Omega_{\mathrm{TVDs}}} < |\nabla u|_{\Omega_{\mathrm{QUICK}}} < |\nabla u|_{\Omega_{\mathrm{CDS}}}$. Since the calculation of sub-grid turbulent viscosity $\nu_{t\Omega}$ depends on the velocity gradient, $\nu_{t\Omega}$ obtained by CDS2 and CDS4 are, as expected, greater than $\nu_{t\Omega}$ from QUICK, TVD-CH, TVD-VL and LL01. As shown in Fig. 7.2f, the total sub-grid turbulent viscosity $\nu_{t\Omega}$ obtained from the dissipative schemes TVD-CH, TVD-VL and LL01 is just half the corresponding turbulent viscosity from CDS2 and CDS4. The distribution of the turbulent viscosity in the tumble plane in Fig. 7.3 reveals the substantial difference between the less and more dissipative numerical schemes, where large values of the turbulent viscosity are found in the simulations using CDS2, CDS4 and QUICK. This

obviously calls into question whether it is reasonable to determine good or bad LES quality based on the sub-grid turbulent viscosity, since the less dissipative numerical schemes would always lead to higher sub-grid turbulent viscosity, hence implying that more turbulence is resolved and less modeled by the more dissipative schemes.

7.9 Turbulent viscosity in dissipative schemes

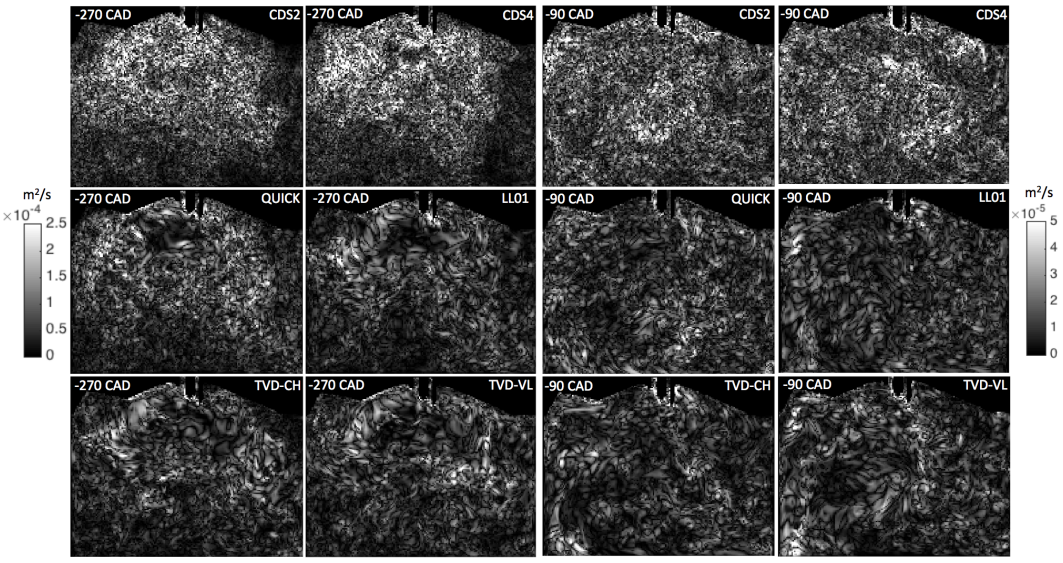

FIGURE 7.3: Distribution of the turbulent viscosity in the tumble plane between different numerical schemes at -270 and -90 CAD from a chosen engine cycle

To examine further the influence of sub-grid modeling in dissipative schemes such as LL01, TVD-CH and TVD-VL, a simulation without sub-grid modeling using LL01 for the discretization of the momentum is carried out. The main reason for this test is that the scheme is very popular in both LES [229, 230] and MILES [231] simulations. The numerical results obtained by two simulations with and without turbulent viscosity can be distinguished in Figure 7.2.

As shown in Fig. 7.2, the differences in the resolved kinetic energy KE_Ω of the in-cylinder flow field as well as in the absolute velocity gradient $|\nabla u|_\Omega$ with and without turbulent viscosity are insignificant. With LL01, the contribution of turbulent viscosity has become minimal and it is hardly noticeable. The turbulent viscosity ν_t depends on the velocity gradients which are better maintained with the CDS or QUICK schemes than with TVD or LL01 schemes. By destroying the velocity gradient, the influence of sub-grid modeling becomes negligible and the engine simulation with LL01 scheme even without using

turbulent viscosity ν_t can still converge and produces similar results as the simulation with the LL01 scheme with a turbulent viscosity ν_t. Figure 7.4 demonstrates clearly the influence of sub-grid modeling on the simulations using CDS2 and LL01. This finding may imply that LES with a limited linear (or similar dissipative) scheme for momentum are actually MILES simulations (implicit LES) with an ineffectual turbulence model. In other words, such simulations might be 'implicitly implicit LES'. In such cases, LES quality criteria based on the turbulent viscosity must fail.

7.10 How much kinetic energy is contained in small flow structures?

Despite the small structures are dominant in the flow field of the simulations using CDS2 and CDS4, the resolved kinetic energy obtained by these simulations is quite similar to the results obtained by TVD-CH, TVD-VL or limited linear schemes (see Fig. 7.2a). This raises a question of how much kinetic energy is contained in small structures. In order to reasonably estimate the amount of kinetic energy containing in these small vortices, we employ a Gaussian filter with the filter size of $3\Delta \times 3\Delta \times 3\Delta$ (Δ denotes the size of the equidistant cells) to filter the small structures out of the flow field. The coefficients of the applied Gaussian filter are presented as a $3 \times 3 \times 3$ dimensional array:

$$\frac{1}{64}\left(\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 2 & 4 & 2 \\ 4 & 8 & 4 \\ 2 & 4 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}\right)$$

The difference in kinetic energy $\Delta \mathrm{KE}_\Omega = \mathrm{KE}_{\mathrm{original}} - \mathrm{KE}_{\mathrm{filtered}}$ between the original flow field and the filtered flow field is a good estimation of how much kinetic energy the small structures contain. Figure 7.5 shows the vertical velocity component and its corresponding high-pass filtered component from different numerical schemes at -180 CAD. The total kinetic energy and the difference $\Delta \mathrm{KE}_\Omega$ between the original and the filtered flow field are plotted in Fig. 7.6. According to our method, the kinetic energy contained in the small flow structures, is less than 10% of the total resolved kinetic energy of the flow field. This finding is interesting because it can explain why

a comparable total kinetic energy is obtained by simulations using different numerical schemes for momentum.

From Figs. 7.1, 7.2 and 7.6, it is clear that the dominantly small structures in the flow field, obtained by simulations using CDS, comprise just a small amount of kinetic energy in comparison to the big structures which govern the flow field of the simulations using TVD or limited linear schemes. In spite of holding a limited amount of kinetic energy, the fluctuations in the flow field significantly contribute to the velocity gradient and eventually, to flame propagation during combustion.

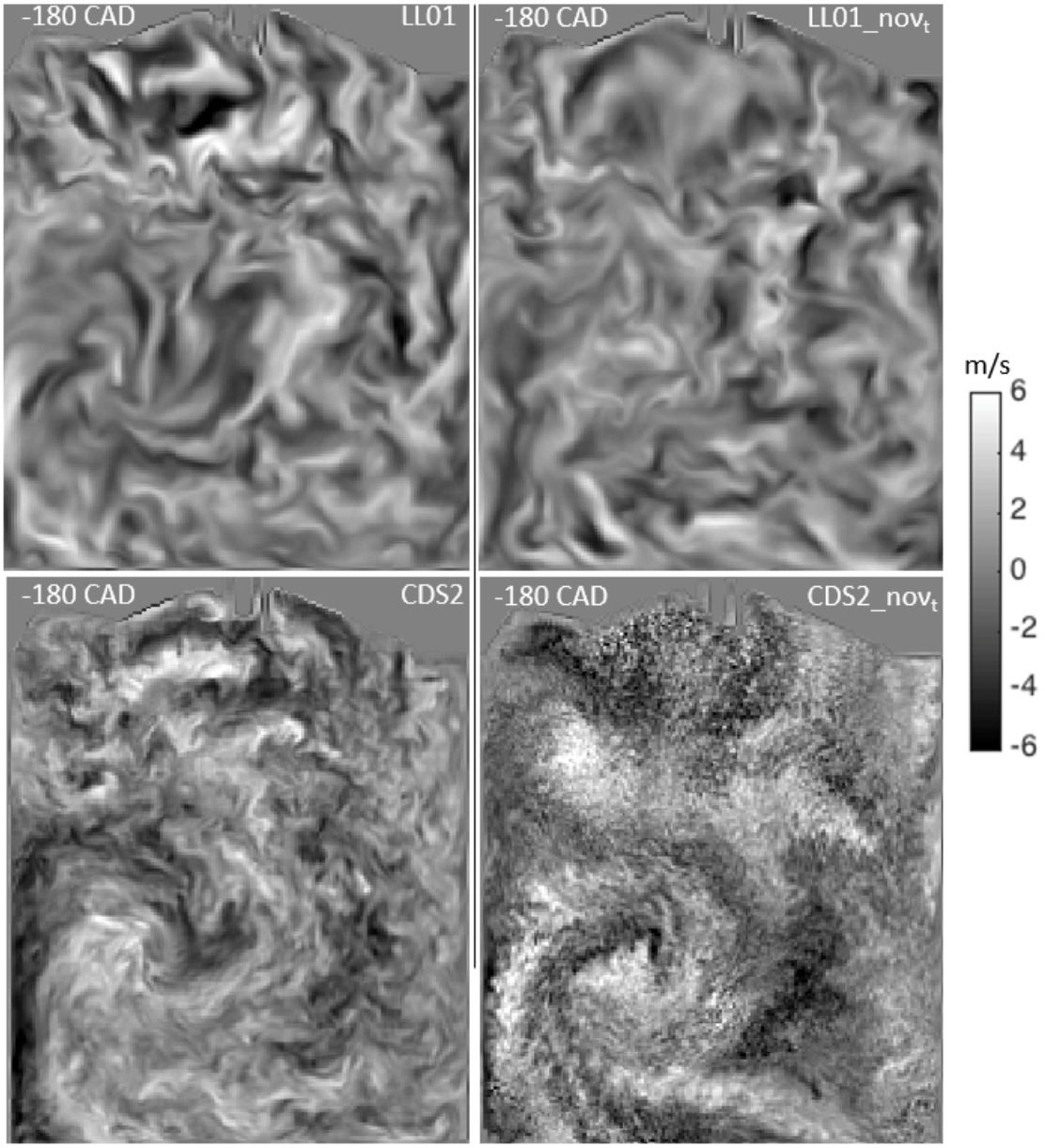

FIGURE 7.4: Image-normal velocity component at -180 CAD obtained from the simulations using CDS2 and LL01 with (left) and without (right) sub-grid modeling.

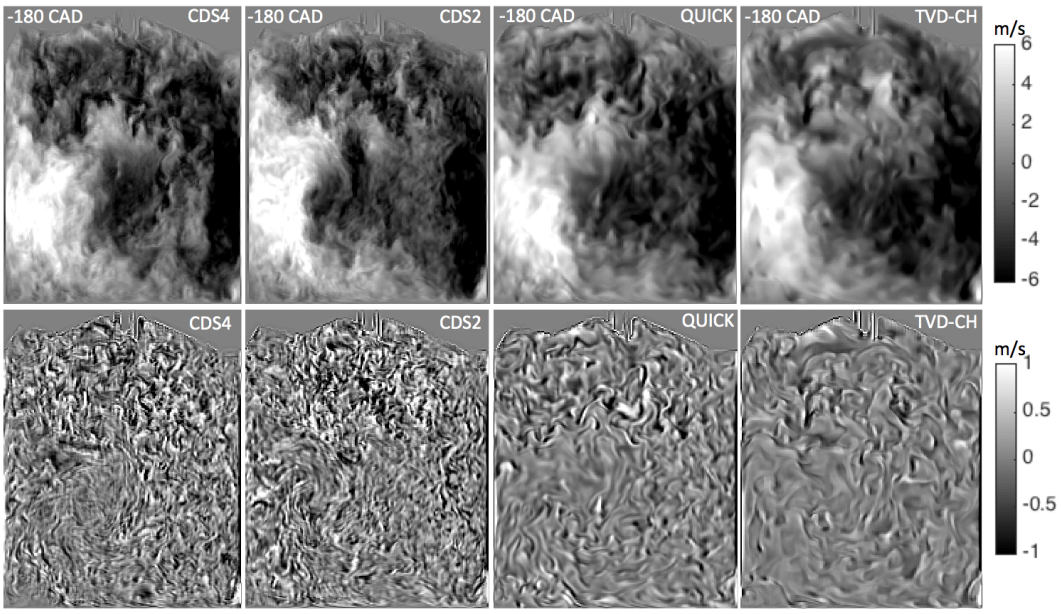

FIGURE 7.5: Illustration of the vertical velocity component (top) and its high pass filtered component (bottom) of different numerical schemes from a chosen engine cycle at -180 CAD using Gaussian filter of size $3\Delta \times 3\Delta \times 3\Delta$.

7.11 Dissipation rate

The resolved dissipation rate is evaluated by the turbulent viscosity ν_t and the phase-averaged strain rate tensor $\langle S_{ij} \rangle$:

$$\epsilon = \nu_t \langle S_{ij} \rangle \langle S_{ij} \rangle \tag{7.6}$$

Figure 7.7 shows the distribution of dissipation rate at -180 CAD in the tumble plane. As expected, the dissipation rate obtained by the simulations with CDS2, CDS4 and QUICK exhibits much higher values in comparison to the ones achieved with TVD-CH, TVD-VL and LL01. The higher dissipation rate resulting in simulations using CDS or QUICK schemes is a quantitative indication of how well the small structures are resolved. It can be clearly seen in Fig. 7.7 that simulations using dissipative schemes (TVD or LL01) can only resolve small structures where strong turbulence exists.

The averaged dissipation rate of the in-cylinder flow for different numerical schemes is shown in Fig. 7.8. The averaged dissipation rate from less dissipative schemes like CDS2 and CDS4 are almost two times larger the dissipation rate resulting from simulations using TVD or LL01 schemes.

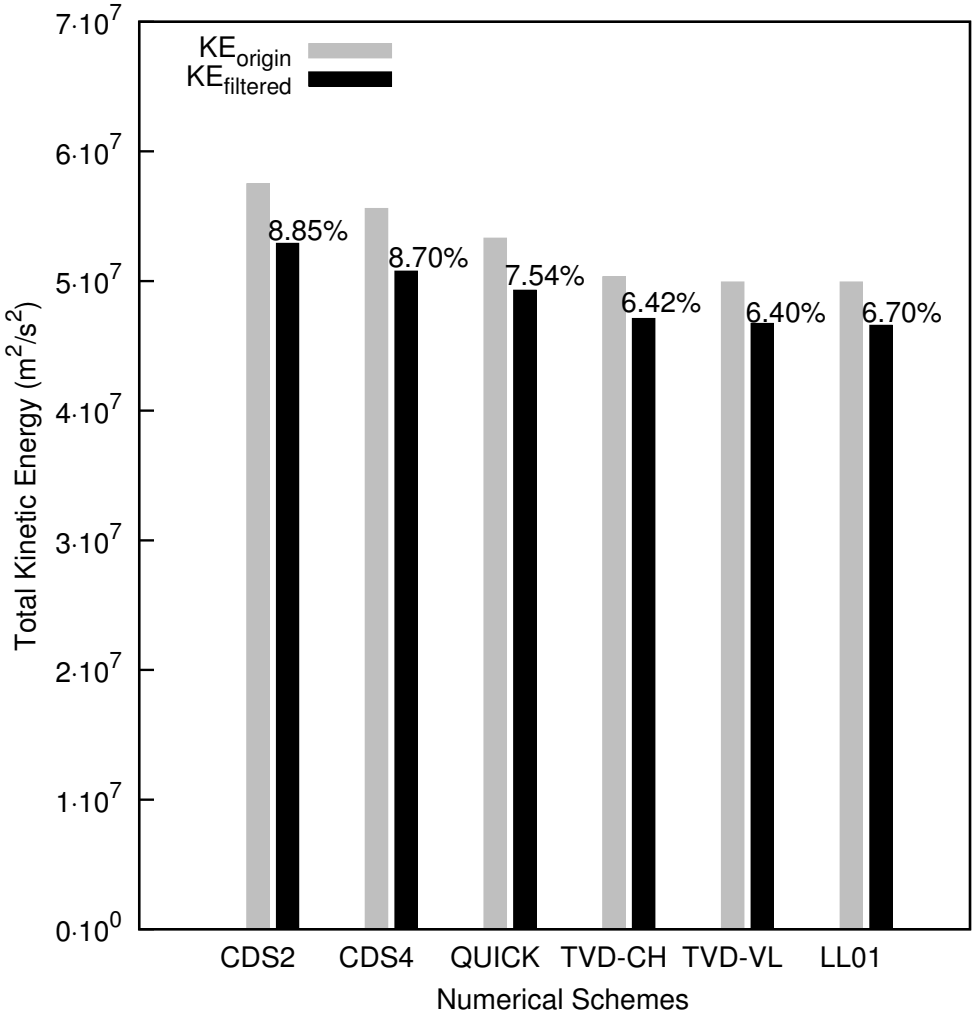

FIGURE 7.6: The difference in total kinetic energy of the original and filtered in-cylinder flow field at -180 CAD

7.12 Estimation of the integral length scale

In order to estimate the integral length scale of the in- cylinder flow, one can compute the spatial correlation function (see Eq. 7.7), in which the two-point correlation is evaluated along two chosen vertical and horizontal sampling lines (see Fig. 7.9). We do not show the integral length-scales themselves as the number of cycles is not sufficient to determine the length-scale with sufficient precision. We show the differences in the correlation functions from which the length scales are determined instead.

The spatial correlation function R of the velocity fluctuation $u'(x)$ reads:

$$\mathrm{R}(r) = \frac{\langle u'(x)u'(x+r)\rangle}{\langle u'(x)^2\rangle} \tag{7.7}$$

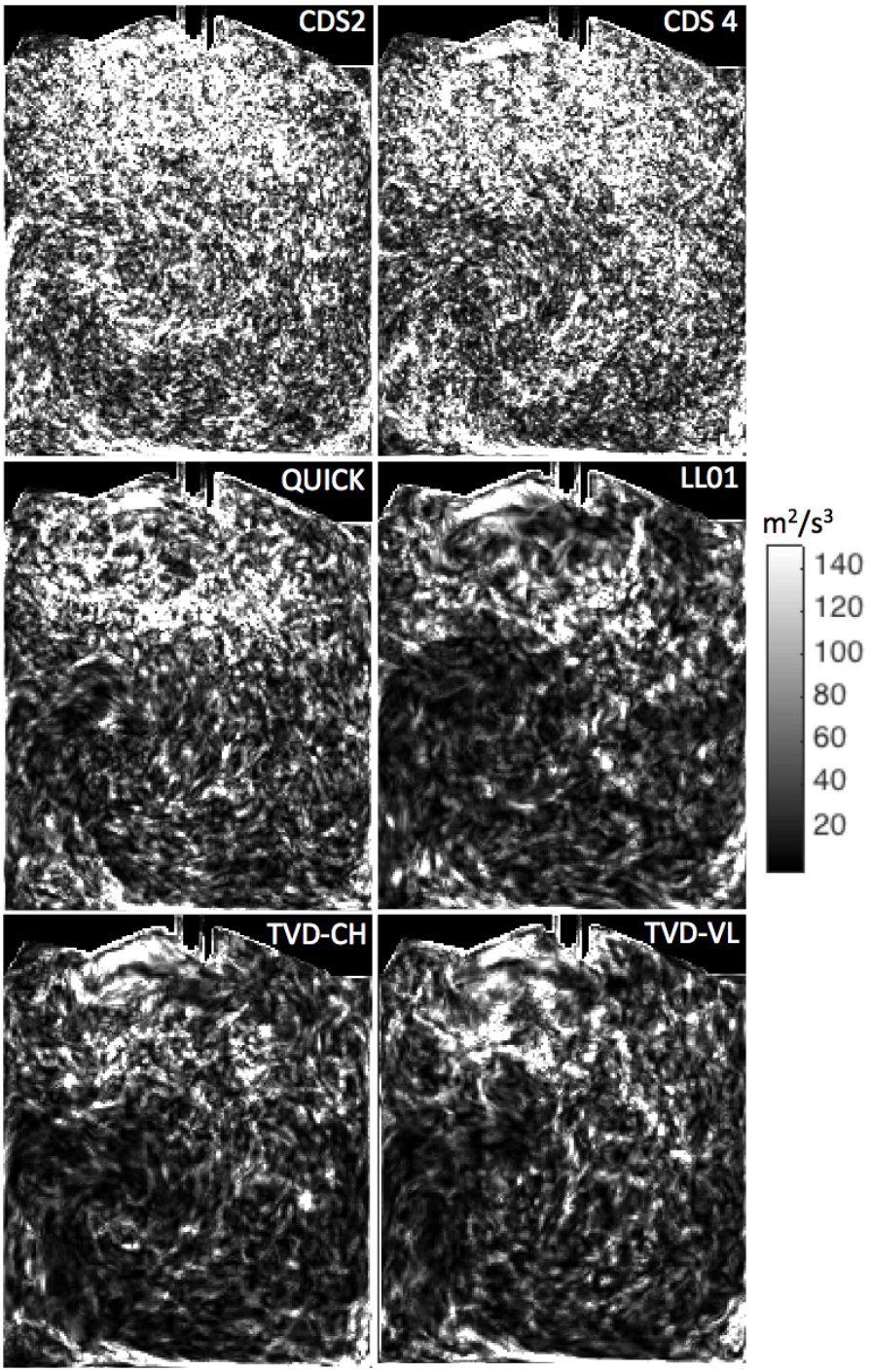

FIGURE 7.7: Dissipation rate obtained at -180 CAD by simulations using different numerical schemes from a chosen engine cycle

The velocity fluctuation $u'(x)$ is calculated from the instantaneous velocity $u(x)$ and the phase averaged velocity $\langle u(x) \rangle$:

$$u'(x) = u(x) - \langle u(x) \rangle \tag{7.8}$$

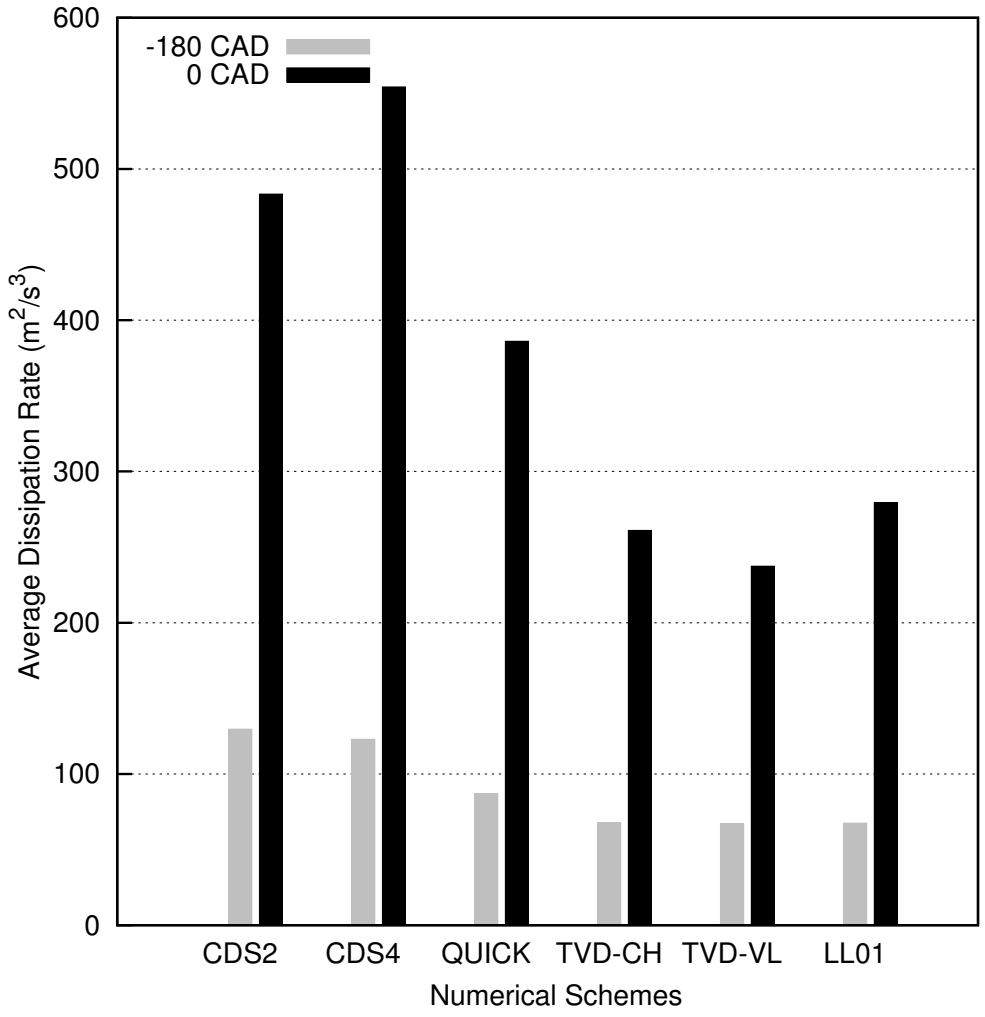

FIGURE 7.8: Averaged dissipation rate obtained at -180 and 0 CAD by simulations using different numerical schemes

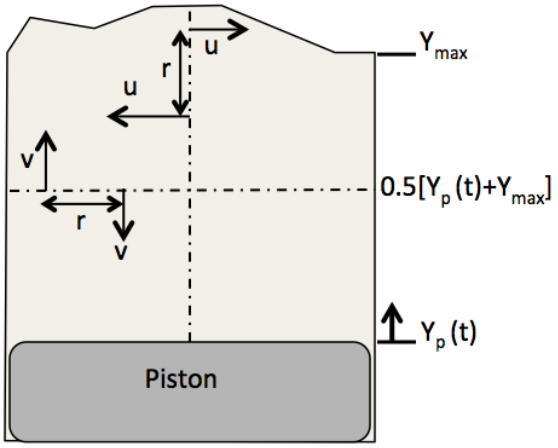

FIGURE 7.9: Vertical and horizontal sampling lines in the tumble plane symmetry for the evaluation of the spatial correlation function

The integral length scale L is calculated by integrating R(r) along the sampling line up to the intersection point P between the curve R (Fig. 7.10) and the horizontal axis (s=OP):

$$\mathrm{L} = \int_0^s \mathrm{R}(r)dr \tag{7.9}$$

As shown in Fig. 7.9, the two-point correlation function is computed along two sampling lines using two velocity components: R_{xx} is computed along the vertical line using the horizontal velocity u and vice versa. The idea is simple, a change in sign of the correlation R_{yy} along the horizontal sampling line or R_{xx} along the vertical sampling line can be considered as a very coarse estimation of the radius of the vortex [142]. More detailed explanation and discussion of the method can be found in the papers [232, 225].

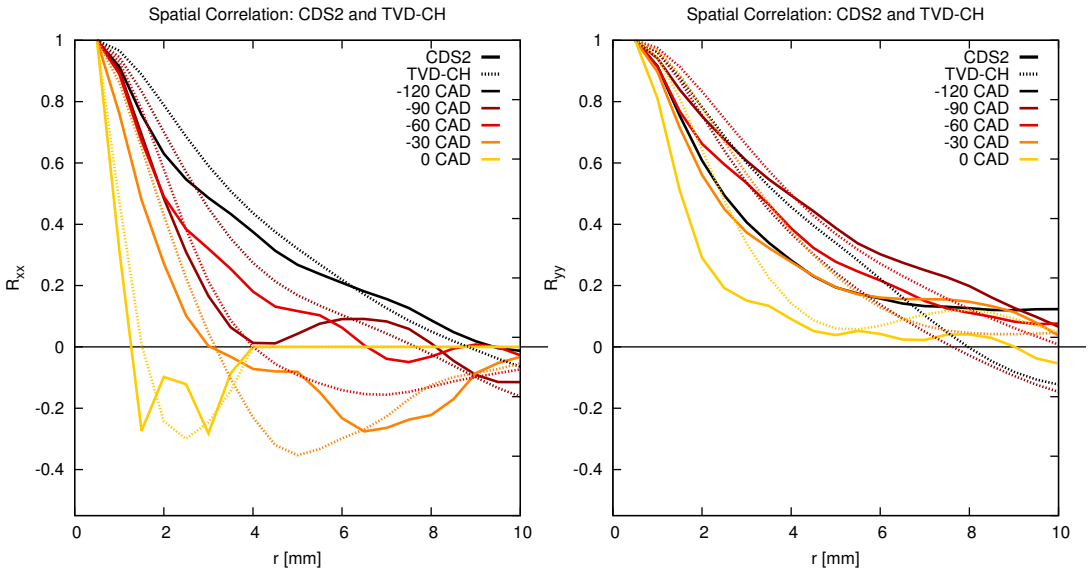

FIGURE 7.10: Two-point correlation coefficients R_{xx} and R_{yy} of the velocity fluctuation along the vertical sampling line using the horizontal velocity fluctuation u' (left) and the horizontal sampling line using the vertical velocity fluctuation v' (right) of the tumble plane

Figure 7.10 shows the calculated two-point correlation function R(r) along two chosen sampling lines. To avoid confusion by plotting too much data (six schemes and 5 crank angle degrees), only results obtained from two simulations with CDS2 and TVD, are presented. Based on the Eq. 7.9 and the comparison of the two-point coefficient curves R_{xx} and R_{yy} between CDS2 and TVD-VL, it would be reasonable to consider that the length scales increase with more dissipative schemes.

7.13 Influence of Numerical schemes on Combustion

Figure 7.12 shows instantaneous volume renderings of the reaction source terms at different crank angle degrees. Since the volume rendering is quite similar between the simulations using CDS2 and CDS4 and between the simulations using TVD-CH, TVD-VL and LL01, we present only the visualization of flame propagation from simulations using CDS2, TVD-CH and QUICK instead of all six simulations. As evidenced by Fig. 7.12, from -8 to 8 CAD, the growth rate of flame area from the CDS2 and QUICK schemes are quite comparable. The flame-wrinkling produced by these less dissipative schemes is much stronger in comparison to the wrinkling from simulations using TVD. Although the total kinetic energy produced by the six numerical schemes is obviously in agreement (Fig. 7.13), the obtained burning rates behave very differently. This is due to the fact, that much of turbulence is still very well 'maintained' by the simulations using CDS in comparison to the ones using TVD schemes, which can be clearly demonstrated by the vertical velocity component at -16 CAD (see Fig. 7.11). Evidently, the integrated turbulent viscosity $\nu_{t\Omega}$ achieved with the simulations using CDS is more than 40% higher than the turbulent viscosity ν_t from the simulations using TVD, which directly results in a larger discrepancy in the integrated reaction source term and the in-cylinder pressure. Interestingly, the difference in $\nu_{t\Omega}$ between the simulation using CDS and QUICK (less than 20%) does not lead to strong deviation in the flame propagation and the in-cylinder pressure, especially between the simulated results from CDS2 and QUICK up to 6 CAD.

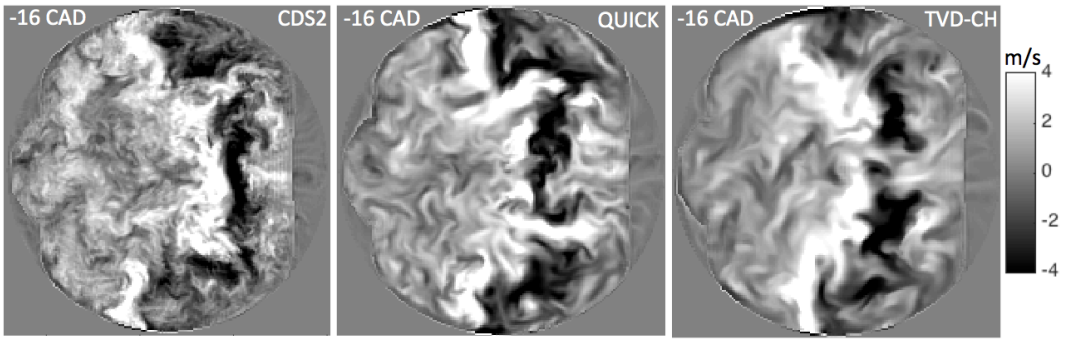

FIGURE 7.11: The vertical velocity obtained by different numerical schemes from a chosen engine cycle at -16 CAD crank angle degree

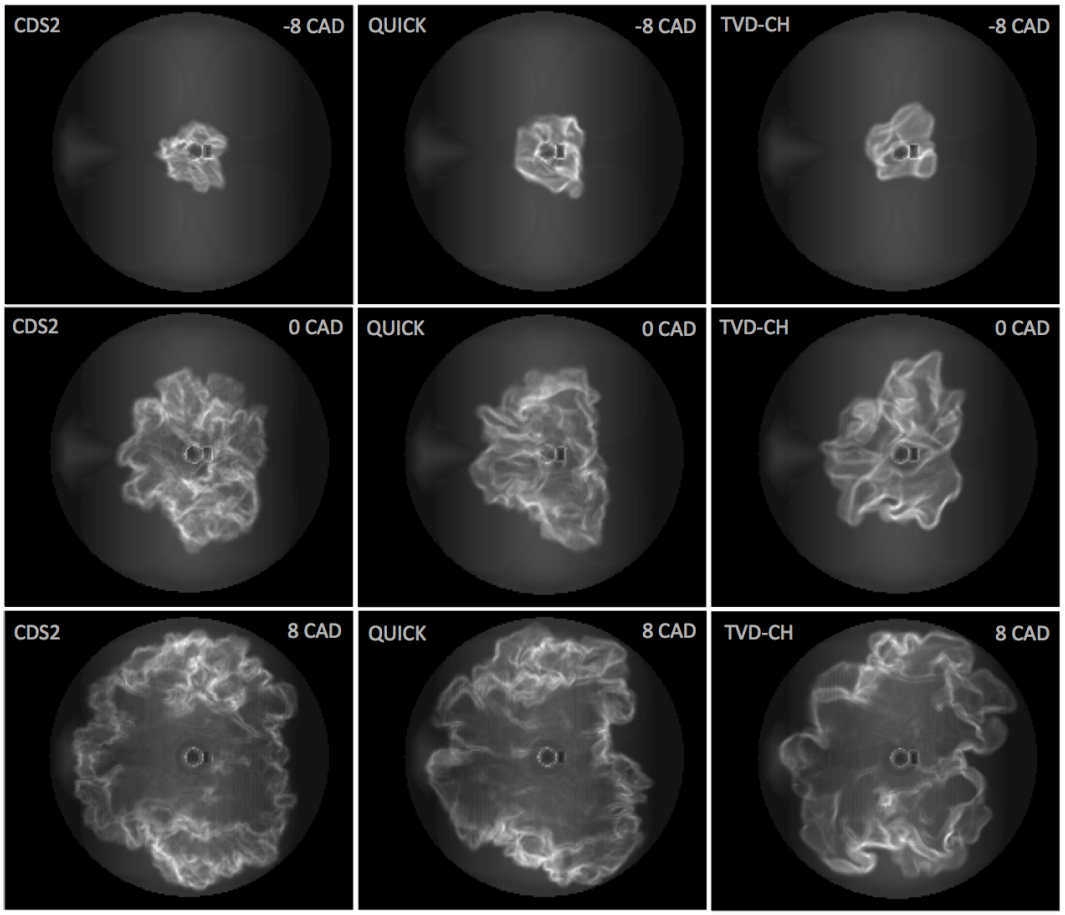

FIGURE 7.12: Volume rendering of reaction source term obtained by different numerical schemes from a chosen engine cycle at different crank angle degree

7.14 LES quality criteria

Determination of good or bad LES quality with a given resolution and numerical scheme has been extensively discussed in many studies [209, 233, 234, 235, 236]. Focusing on engine simulations, some authors [225, 142] have applied several LES quality indices [209, 74]. These LES indices are used to estimate, whether a given resolution is sufficient for a 'good LES'. However, most of these criteria are likely to fail due to the influence of numerical schemes on the numerical results. We have shown that simulations using highly dissipative schemes could be considered as a 'good LES' since they seem to achieve less unresolved turbulent kinetic energy or less turbulent viscosity in comparison to the simulations using less dissipative schemes such as CDS2, CDS4 or QUICK. Similar arguments can also be found in the work of others [233, 173]. Obviously, any LES quality criteria, that consider the minimum of turbulent viscosity or unresolved turbulent kinetic energy as an indicator for a 'good LES', should be taken with care.

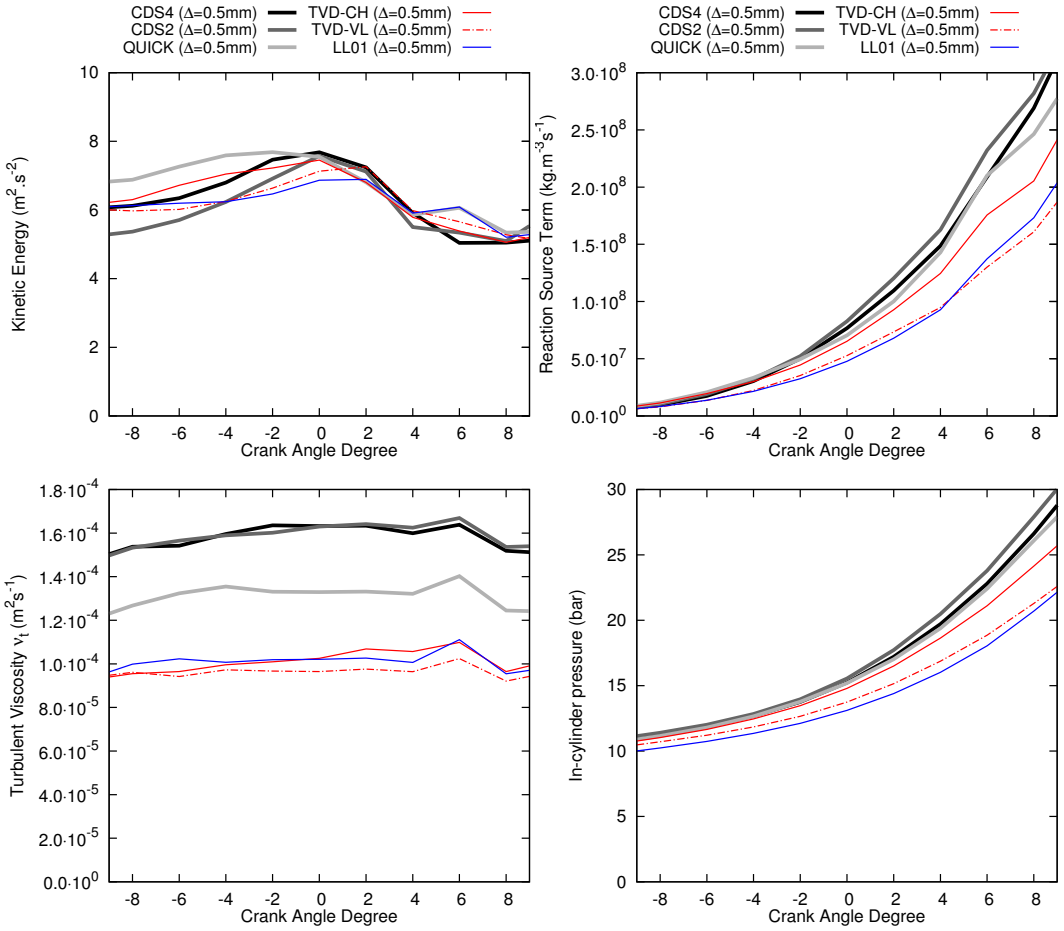

FIGURE 7.13: Comparison of the integrated kinetic energy, integrated turbulent viscosity, integrated reaction source term and in-cylinder pressure obtained for different numerical schemes

To demonstrate the inadequacy of such LES quality criteria, we apply them to the LES engine simulations on the same grid using different numerical schemes.

The first criterion is the ratio of resolved turbulent kinetic energy $k_{res}(x,t)$ to the amount of total turbulent kinetic energy $(k_{res}(x,t) + k_{sgs}(x,t))$.

$$\mathrm{M(x,t)} = \frac{k_{res}(x,t)}{k_{res}(x,t) + k_{sgs}(x,t)} \tag{7.10}$$

If a LES computation is able to resolve more than 80% of the total turbulent kinetic energy (M(x,t)>0.8), then it is supposed to be a 'good LES'. The calculations of resolved

kinetic energy $k_{res}(x,t)$ are outlined below.

$$k_{res}(x,t) = \frac{1}{2}\big(U_i - \langle U_i \rangle\big)^2 \tag{7.11}$$

where $\langle U_i \rangle$ is the phase-averaged velocity component. The sub-grid kinetic energy $k_{sgs}(x,t)$ can be modeled from the phase averaged strain rate tensor $|\langle S_{ij} \rangle|^2$ according to Eq. 7.12 [237, 225] with the constant C_i set to 0.202:

$$k_{sgs}(x,t) = 2C_i\Delta^2|\langle S_{ij} \rangle|^2 \tag{7.12}$$

Comment: it should be noted that the Eq. 7.12 is used to calculate the turbulent kinetic energy for the evaluation of the energy criterion M(x,t) *, which follows closely the work of di Mare et al. [225]. For the modeling of combustion, the model proposed by Muppala et al. [32] and the turbulent kinetic energy k_{sgs} from [107] are employed in this study.*

Figure 7.14 shows the distributed value of M(x, t) as computed from the in-cylinder flow at -270, -180, -90 CAD. As discussed, six investigated schemes lead to results that satisfy the 'good LES criterion' related to resolved kinetic energy. Obviously, from the PDF of M(x,t) in Fig. 7.14, the number of points exceeding the value of 0.95 by numerical schemes like TVDs and QUICK is greater than for CDS2 and CDS4. This implies that the less dissipative schemes like CDS and CDS4 resolve a smaller part of the turbulent kinetic energy in comparison to TVD, which is clearly not true.

A second common criterion for LES quality is based on the ratio of the effective viscosity $(\overline{\nu} + \overline{\nu}_t)$ to the molecular viscosity $\overline{\nu}$. A good LES is supposed to obtain a value of IQ_ν above 0.8, while an IQ_ν value of 0.952, is considered a fully resolved DNS [209, 173].

$$IQ_\nu = (1.0 + 0.05(\frac{\overline{\nu} + \overline{\nu}_t}{\overline{\nu}})^{0.53})^{-1} \tag{7.13}$$

According to the PDF of LES IQ_ν at -270, -180 and -90 CAD in Fig. 7.14, six engine simulations with different numerical schemes satisfy the LES IQ_ν index with the value of IQ_ν above 0.8 for all the computational cells. Similar to the energy criterion, a larger number of computational cells from simulations using dissipative schemes such as TVD and LL01 achieves a much higher value of LES IQ_ν in comparison to the LES IQ_ν obtained by simulations with CDS and QUICK. Figure 7.15 shows the averaged values of the two LES quality criteria, in which simulations using TVD schemes always show

a better 'LES quality' than the simulations with CDS or QUICK schemes. As stressed in sections 7.8 and 7.9, it is not a good approach to use a quantity, that is strongly influenced by the numerical schemes, to evaluate the quality of a simulation.

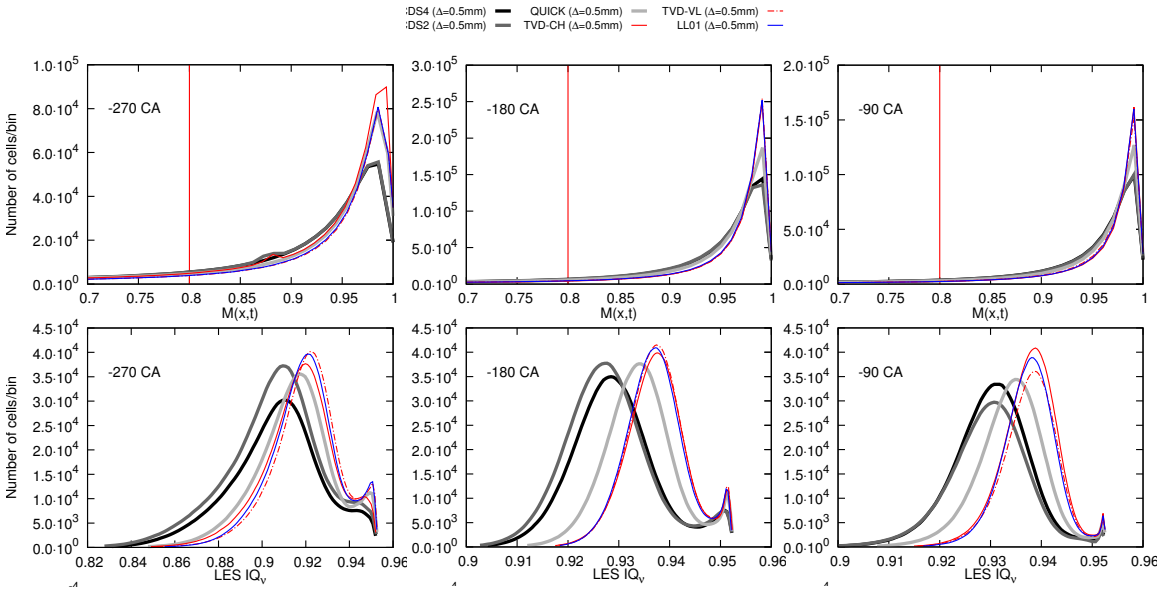

FIGURE 7.14: PDF of two LES quality indices which are evaluated for the in-cylinder flow with a bin width of 0.002

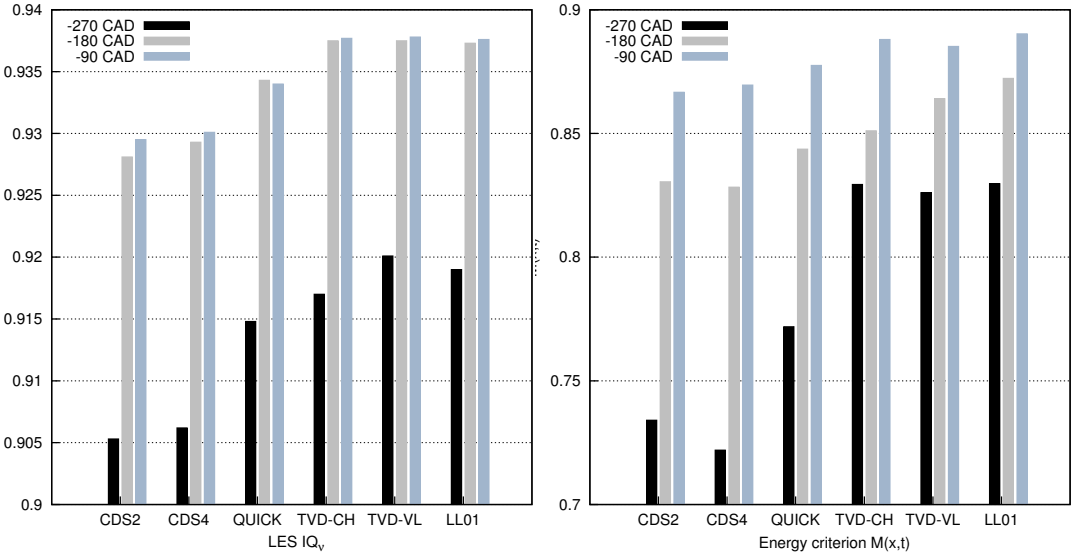

FIGURE 7.15: Averaged value of LES IQ (left) and the ratio between the resolved and the total kinetic energy M(x,t) (right).

7.15 Conclusion

This work studies the numerical effects on the flow behavior as well as the prediction of flame propagation in the context of engine simulation using LES. To carry out the investigation, six different numerical schemes (CDS2, CDS4, QUICK, TVD-CH, TVD-VL, Limited Linear) were used to discretize the momentum equation over multiple engine cycles using a real engine geometry [141].

As it was expected, small structures were found to be dominant in the flow field from simulations using less dissipative schemes (CDS2, CDS4 and QUICK), while big and smooth structures prevailed in the flow field from simulations using more dissipative schemes (TVDs and limited linear). Interestingly, it is found that the total kinetic energy of the in-cylinder flow of all the simulations (using less or more dissipative schemes) was comparable from intake to exhaust stroke, despite much difference observed in the flow fields.

In this study, we have shown that the contribution of sub-grid modeling may be negligible for simulations using dissipative schemes such as limited linear or TVD. Since the velocity gradient is not well-resolved by the more dissipative schemes, the calculated turbulent viscosity is much smaller than the one obtained by simulations using less dissipative schemes.

Our study also showed that the numerical schemes have a strong effect on the combustion process since flame propagation is strongly influenced by the level of turbulence that must be well-maintained throughout the cycle. Obtaining a correct wrinkling factor or flame propagation requires a well-resolved flow field with sufficient turbulence.

We also confirm that many common LES quality criteria are inadequate to determine a ‘good LES’. We have illustrated that simulations using highly dissipative schemes are likely to be considered ‘good LES’ under these criteria, which is clearly not correct.

Chapter 8

Conclusions and outlook

This chapter summarizes the findings and the development of the engine simulation using LES, that was conducted for this PhD thesis. Besides the major advantages achieved, the current work still has some drawbacks that need to be outlined. Therefore, potential improvements and recommendations for future development are also discussed.

8.1 Conclusion

This thesis focused on the numerical simulations of the internal combustion engines using a particle-based method and large eddy simulation. The main aim of this work was to develop the numerical treatments that can be effectively applied to simulate the fluid dynamics and the combustion that occur inside an ICE. Several issues in the context of engine simulations were considered: (I) a simple strategy for mesh-generation for the complex engine geometries; (II) efficient numerical treatments to model the moving boundaries in order to avoid dynamic mesh adaptation during the simulation; (III) an LES Navier-Stokes solver for a fully compressible fluid; (IV) suitable mathematical models for turbulent combustion in internal combustion engines.

In the scope of this work, equidistant Cartesian grids were used for the computational domain, which can be easily and quickly generated for the complex engine geometries using available voxelization tools. The new approach to model the moving boundaries without dynamic meshing was introduced, in which the moving objects inside a flow field are numerically described as a cloud of Lagrangian particles. The motion of the solid

body is therefore handled by solving a system of ordinary differential equations, which is a computationally efficient approach (in general, one can also specify the positions of the particles without solving the ODEs if the positions of the particles are known). The flow conditions at the moving boundaries are numerically treated using the velocity, the acceleration of the moving particles and the immersed boundary method. To account for the fluid dynamics inside the engines, the filtered Navier-Stokes equations were solved for a fully compressible flow using the finite volume method. Turbulent combustion was modelled based on the flame surface density approach.

The approach proposed in this thesis offers a convenient way to perform multi-cycle engine simulations using a single mesh. On large scale simulations, this approach has proven its suitability for high performance computing for the following reasons: (1) because of the simplicity in mesh generation, a computational grid of a hundred million cells can be generated within an hour; (2) the grid decomposition for parallelization is independent from geometrical features; (3) the numerical treatments for the moving boundaries using Lagrangian particles and immersed boundary can be efficiently executed in parallel. As it was shown in Chapter 6, a good scalability of the computations using a few thousand processors was achieved by the proposed approach. Also in Chapter 6, the efficiency of parallelization of the method was demonstrated for an engine simulation on a domain of 180 million cells using 8196 CPUs.

In this work, accurate and less dissipative numerical schemes on an equidistant Cartesian grid were employed, so that the turbulence of the in-cylinder flow is resolved and maintained, and the flame propagation can be well-captured throughout the engine simulation. The method was validated by multi-cycle LES simulations of the motored-case and the fired-case using different engine geometries, like the single-centered valve engine or the complex geometries of 4-stroke engines (such as those operated by research groups in the University of Duisburg-Essen and the Technical University of Darmstadt). The obtained results such as the velocity, the velocity fluctuations, the in-cylinder pressure and the flame propagation were compared against the corresponding experimental data as well as other numerical results. Good agreement between the simulations and the experiments was achieved, demonstrating the correctness of the proposed method.

Additionally, investigation of the numerical effects on the LES of ICE was conducted. This study showed that the large structures in the flow field can be well-captured by

a wide range of chosen numerical schemes, in which a similar amount of total resolved kinetic energy was obtained. Although the small structures in turbulent flow may contain no more than 10% of the total resolved kinetic energy, they play a significant role in the combustion processes inside the engine cylinder. Maintaining this level of turbulence in a long process from the intake, compression and ignition is essential for numerical simulations of the engine. It is shown that many popular numerical schemes dissipate the small structures rather quickly, which results in a low wrinkling factor and hence turbulent flame speed.

8.2 Outlook

Numerical simulations of internal combustion engines are complicated in different aspects, from the mathematical modelling to the numerical treatments. In the current work, there are certain gaps that need to be filled, some drawbacks that need to be addressed and some potential aspects that can be further improved in the future work.

First, as the number of fluid- and solid cells changes during the engine simulation using IBM, the ratio of solid to fluid cells can be in the range of 1.1 to 2.6 (see Chapter 6). For instance, when the piston reaches the TDC, the number of fluid cells reduces to less than 30% of the total number of computational cells in the domain. Significant amount of computational resources is unnecessarily spent on solving Navier-Stokes equations for the solid cells. To circumvent this problem, the approach of Mittal et al. [202] can be used to reduce the unnecessary computations. In this approach, the domain is decomposed into a number of sub-domains that is larger than the number of processors. The sub-domains consisting of solid cells only will not be considered and the remaining sub-domains will be distributed amongst the given number of processors. The Reduced Cartesian Communicator (RCC), which belongs to the Message Passing Interface (MPI) library, is employed to exchange the data between these fluid regions. This approach from Mittal et al. was proven to be efficient due to the reduction of a significant number of solid cells which are unnecessarily used to solve the Navier-Stokes equations.

Second, the proposed approach employed the equidistant Cartesian grid to numerically describe the engine geometry. Without local refinement, to resolve some geometrical

features like the valve gaps, the crevice volume or the spark-plug would require a significantly large number of computational cells. Therefore, engine simulations using an equidistant Cartesian grid will always require more computational cells than simulations using unstructured grids. A suitable local refinement would tremendously reduce the number of the computational cells and allow a more accurate description of the engine's geometrical features. Additionally, local refinement will allow the boundary layer to be resolved, which is important for simulating the flow in a confined domain - such as an engine. The hierarchical refinement approach of Quirk [238] or the local adaptive mesh refinement proposed by Berger et al. [239] can provide a solution for mesh refinement in a Cartesian grid. The idea is simple: in the area that needs to be refined, another layer of a finer Cartesian grid will be added. The Navier-Stokes equations will then be solved in both the coarse- and the fine grids. It must be noted that the boundary conditions of the simulation on the fine grid are directly taken from the numerical results obtained on the coarse grid. Following this, the results from the fine grid will be mapped onto the corresponding area of the coarse grid. Both approaches are straightforward to implement without changing the existing flow solver and the particle-based technique for the moving boundaries.

Third, in the current work, the interaction between turbulence and chemistry was modelled by an FSD model with a single-step mechanism for the heat release. It is clear that the chosen approach is simple, avoiding additional complexity to the flow simulation inside an engine. However, the employed models may not be adequate to reproduce the complexity of chemical reactions, including the stretch effects on the flame or the complex chemistry. As it was shown in the fired-motor in Chapter 6, despite capturing the flame propagation well, the simulation was not able to correctly resolve the peak in the in-cylinder pressure during engine operation. It is clear that a single-step mechanism is not adequate to reproduce the complexity of the chemical reactions inside the engine combustion chamber. A proper description of chemical reactions is therefore necessary. Using tabulated chemistry such as FGM [20] or IDLM [17, 18] in combination with the mathematical models like FSD [28, 29] or ATF [33, 38] for turbulent flame would further improve the numerical prediction for engine simulations.

Bibliography

[1] D. Sperling and D. Gordon. *Two billion cars: driving toward sustainability*. Oxford University Press, 2009.

[2] EU. *European emission standards*. https://en.wikipedia.org/wiki/European_emission_standards. Accessed: 2017-03-01.

[3] F. Bracco et al. *Two-phase, two-dimensional, unsteady combustion in internal combustion engines; theoretical-experimental results*. Tech. rep. SAE Technical Paper, 1976.

[4] A. Boni, M. Chapman, and G. Schneyer. "Computer simulation of combustion processes in a stratified charge engine". In: *Acta Astronautica* 3.3-4 (1976), pp. 293–307.

[5] R. Diwakar. "Interaction of combustion with aerodynamic flowfield in an internal combustion reciprocationg engine: a two-dimensional numerical solution". PhD thesis. University of Maryland, College Park., 1977.

[6] M. Griffin et al. "Computational fluid dynamics applied to flows in an internal combustion engine". In: *16th Aerospace Sciences Meeting*. 1978, p. 57.

[7] M. Chapman. *Two dimensional numerical simulation of inlet manifold flow in a four cylinder internal combustion engine*. Tech. rep. SAE Technical Paper, 1979.

[8] M. Takizawa et al. *A study of gas exchange process simulation of an automotive multi-cylinder internal combustion engine*. Tech. rep. SAE Technical Paper, 1982.

[9] A. Amsden et al. *KIVA: A computer program for two-and three-dimensional fluid flows with chemical reactions and fuel sprays*. Tech. rep. Los Alamos National Lab., NM (USA), 1985.

[10] B. Khalighi et al. *Computation and measurement of flow and combustion in a four-valve engine with intake variations*. Tech. rep. SAE Technical Paper, 1995.

[11] T. Butler et al. "Multidimensional numerical simulation of reactive flow in internal combustion engines". In: *Progress in Energy and Combustion Science* 7.4 (1981), pp. 293–315.

[12] K. Naitoh et al. *Large eddy simulation of premixed-flame in engine based on the multi-level formulation and the renormalization group theory.* Tech. rep. SAE Technical Paper, 1992.

[13] Z. Han and R. D. Reitz. "Turbulence modeling of internal combustion engines using RNG κ-ε models". In: *Combustion science and technology* 106.4-6 (1995), pp. 267–295.

[14] I. Celik et al. "Prediction of in-cylinder turbulence for IC engines". In: *Combustion science and technology* 153.1 (2000), pp. 339–368.

[15] I. Celik, I. Yavuz, and A. Smirnov. "Large eddy simulations of in-cylinder turbulence for internal combustion engines: a review". In: *International Journal of Engine Research* 2.2 (2001), pp. 119–148.

[16] S. Richard et al. "Towards large eddy simulation of combustion in spark ignition engines". In: *Proceedings of the Combustion Institute* 31.2 (2007), pp. 3059–3066.

[17] U. Maas and S. B. Pope. "Implementation of simplified chemical kinetics based on intrinsic low-dimensional manifolds". In: *Symposium (International) on Combustion.* Vol. 24. 1. Elsevier. 1992, pp. 103–112.

[18] U. Maas and S. B. Pope. "Simplifying chemical kinetics: intrinsic low-dimensional manifolds in composition space". In: *Combustion and flame* 88.3 (1992), pp. 239–264.

[19] O. Gicquel, N. Darabiha, and D. Thévenin. "Liminar premixed hydrogen/air counterflow flame simulations using flame prolongation of ILDM with differential diffusion". In: *Proceedings of the Combustion Institute* 28.2 (2000), pp. 1901–1908.

[20] J. Van Oijen, F. Lammers, and L. De Goey. "Modeling of complex premixed burner systems by using flamelet-generated manifolds". In: *Combustion and Flame* 127.3 (2001), pp. 2124–2134.

[21] S. B. Pope. "Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation". In: (1997).

[22] B. Yang and S. Pope. "Treating chemistry in combustion with detailed mechanisms in situ adaptive tabulation in principal directions premixed combustion". In: *Combustion and Flame* 112.1 (1998), pp. 85–112.

[23] A. R. Kerstein, W. T. Ashurst, and F. A. Williams. "Field equation for interface propagation in an unsteady homogeneous flow field". In: *Physical Review A* 37.7 (1988), p. 2728.

[24] N. Peters and M. Dekena. "Combustion modeling with the G-equation". In: *Oil & Gas Science and Technology* 54.2 (1999), pp. 265–270.

[25] M. Oberlack, H. Wenzel, and N. Peters. "Symmetries and averaging of the G-equation for premixed combustion". In: *TSFP Digital Library Online*. Begel House Inc. 2001.

[26] E. Schneider et al. "Study on the potential of BML-approach and G-equation concept-based models for predicting swirling partially premixed combustion systems: URANS computations". In: *Combustion and Flame* 152.4 (2008), pp. 548–572.

[27] Z. Tan, S.-C. Kong, and R. D. Reitz. *Modeling premixed and direct injection SI engine combustion using the G-equation model*. Tech. rep. SAE Technical Paper, 2003.

[28] F. Marble and J. Broadwell. "The coherent flame model of non-premixed turbulent combustion". In: *Project Squid TRW-9-PU, Project Squid Headquarters, Chaffee Hall, Purdue University* (1977).

[29] S. Pope. "The evolution of surfaces in turbulence". In: *International journal of engineering science* 26.5 (1988), pp. 445–469.

[30] R. Cant, S. Pope, and K. Bray. "Modelling of flamelet surface-to-volume ratio in turbulent premixed combustion". In: *Symposium (International) on Combustion*. Vol. 23. 1. Elsevier. 1991, pp. 809–815.

[31] K. Bray, M. Champion, and P. A. Libby. "The interaction between turbulence and chemistry in premixed turbulent flames". In: *Turbulent Reactive Flows*. Springer, 1989, pp. 541–563.

[32] S. Muppala et al. "Development of an algebraic reaction rate closure for the numerical calculation of turbulent premixed methane, ethylene, and propane/air flames for pressures up to 1.0 MPa". In: *Combustion and flame* 140.4 (2005), pp. 257–266.

[33] T. Butler and P. O'rourke. "A numerical method for two dimensional unsteady reacting flows". In: *Symposium (International) on Combustion*. Vol. 16. 1. Elsevier. 1977, pp. 1503–1515.

[34] O. Colin et al. "A thickened flame model for large eddy simulations of turbulent premixed combustion". In: *Physics of Fluids* 12.7 (2000), pp. 1843–1863.

[35] G. Kuenne, A. Ketelheun, and J. Janicka. "LES modeling of premixed combustion using a thickened flame approach coupled with FGM tabulated chemistry". In: *Combustion and Flame* 158.9 (2011), pp. 1750–1767.

[36] F. Proch and A. Kempf. "Modeling heat loss effects in the large eddy simulation of a model gas turbine combustor with premixed flamelet generated manifolds". In: *Proceedings of the Combustion Institute* 35.3 (2015), pp. 3337–3345.

[37] G. Wang, M. Boileau, and D. Veynante. "Implementation of a dynamic thickened flame model for large eddy simulations of turbulent premixed combustion". In: *Combustion and Flame* 158.11 (2011), pp. 2199–2213.

[38] F. Charlette, C. Meneveau, and D. Veynante. "A power-law flame wrinkling model for LES of premixed turbulent combustion Part I: Non-dynamic formulation and initial tests". In: *Combustion and Flame* 131.1 (2002), pp. 159–180.

[39] R. Bilger. "Conditional moment closure for turbulent reacting flow". In: *Physics of Fluids A: Fluid Dynamics* 5.2 (1993), pp. 436–444.

[40] E. Mastorakos and R. Bilger. "Second-order conditional moment closure for the autoignition of turbulent flows". In: *Physics of Fluids* 10.6 (1998), pp. 1246–1248.

[41] A. Y. Klimenko and R. W. Bilger. "Conditional moment closure for turbulent combustion". In: *Progress in energy and combustion science* 25.6 (1999), pp. 595–687.

[42] W. K. Bushe and H. Steiner. "Conditional moment closure for large eddy simulation of nonpremixed turbulent reacting flows". In: *Physics of Fluids* 11.7 (1999), pp. 1896–1906.

[43] S. H. Kim, K. Y. Huh, and L. Tao. "Application of the elliptic conditional moment closure model to a two-dimensional nonpremixed methanol bluff-body flame". In: *Combustion and Flame* 120.1 (2000), pp. 75–90.

[44] C. M. Cha, G. Kosály, and H. Pitsch. "Modeling extinction and reignition in turbulent nonpremixed combustion using a doubly-conditional moment closure approach". In: *Physics of Fluids* 13.12 (2001), pp. 3824–3834.

[45] V. Moureau et al. *Towards large eddy simulation in internal-combustion engines: simulation of a compressed tumble flow*. Tech. rep. SAE Technical Paper, 2004.

[46] O. Vermorel et al. *Multi-cycle LES simulations of flow and combustion in a PFI SI 4-valve production engine*. Tech. rep. SAE Technical Paper, 2007.

[47] L. Thobois, R. Lauvergne, and T. Poinsot. *Using LES to investigate reacting flow physics in engine design process*. Tech. rep. SAE Technical Paper, 2007.

[48] B. Enaux et al. "Large-eddy simulation of a motored single-cylinder piston engine: numerical strategies and validation". In: *Flow, turbulence and combustion* 86.2 (2011), pp. 153–177.

[49] B. Hu and C. J. Rutland. *Flamelet modeling with LES for diesel engine simulations*. Tech. rep. SAE Technical Paper, 2006.

[50] G. Lecocq et al. "A new LES model coupling flame surface density and tabulated kinetics approaches to investigate knock and pre-ignition in piston engines". In: *Proceedings of the Combustion Institute* 33.2 (2011), pp. 3105–3114.

[51] M. Patterson et al. *Modeling the effects of fuel injection characteristics on diesel engine soot and NOx emissions*. Tech. rep. SAE Technical Paper, 1994.

[52] J. Duclos, M. Zolver, and T. Baritaud. "3D modeling of combustion for DI-SI engines". In: *Oil & Gas Science and Technology* 54.2 (1999), pp. 259–264.

[53] G. Eggenspieler and S. Menon. "Large-eddy simulation of pollutant emission in a DOE-HAT combustor". In: *Journal of Propulsion and Power* 20.6 (2004), pp. 1076–1085.

[54] G. Eggenspieler and S. Menon. "Combustion and emission modelling near lean blow-out in a gas turbine engine". In: *Progress in Computational Fluid Dynamics, an International Journal* 5.6 (2005), pp. 281–297.

[55] W. Lazik et al. "Development of lean-burn low-NOx combustion technology at Rolls-Royce Deutschland". In: *ASME Turbo Expo 2008: Power for Land, Sea, and Air*. American Society of Mechanical Engineers. 2008, pp. 797–807.

[56] S. Banerjee et al. *Validation of an LES multi mode combustion model for diesel combustion*. Tech. rep. SAE Technical Paper, 2010.

[57] S. Fontanesi et al. "Knock tendency prediction in a high performance engine using LES and tabulated chemistry". In: *SAE International Journal of Fuels and Lubricants* 6.1 (2013), pp. 98–118.

[58] I. Yavuz, I. Celik, and M. McMillian. *Knock Prediction in Reciprocating Gas-Engines Using Detailed Chemical Kinetics*. Tech. rep. SAE Technical Paper, 2001.

[59] X. Zhen, Y. Wang, and Y. Zhu. "Study of knock in a high compression ratio SI methanol engine using LES with detailed chemical kinetics". In: *Energy Conversion and Management* 75 (2013), pp. 523–531.

[60] S. Fontanesi, A. d'Adamo, and C. J. Rutland. "Large-eddy simulation analysis of spark configuration effect on cycle-to-cycle variability of combustion and knock". In: *International Journal of Engine Research* 16.3 (2015), pp. 403–418.

[61] O. Vermorel et al. "Towards the understanding of cyclic variability in a spark ignited engine using multi-cycle LES". In: *Combustion and Flame* 156.8 (2009), pp. 1525–1541.

[62] B. Enaux et al. "LES study of cycle-to-cycle variations in a spark ignition engine". In: *Proceedings of the combustion Institute* 33.2 (2011), pp. 3115–3122.

[63] C. Pera, S. Richard, and C. Angelberger. *Exploitation of multi-cycle engine LES to introduce physical perturbations in 1D engine models for reproducing CCV*. Tech. rep. SAE Technical Paper, 2012.

[64] B. Roux et al. "Prediction of cyclic combustion variability in internal combustion engines via coupled 1D-3D LES method". In: *Proceedings of the 6th European Conference on Computational Fluid Dynamics*. 2014, pp. 5474–5485.

[65] A. d'Adamo et al. "LES modelling of spark-ignition cycle-to-cycle variability on a highly downsized DISI engine". In: *SAE International Journal of Engines* 8.2015-24-2403 (2015), pp. 2029–2041.

[66] V. Granet et al. "Large-eddy simulation and experimental study of cycle-to-cycle variations of stable and unstable operating points in a spark ignition engine". In: *Combustion and Flame* 159.4 (2012), pp. 1562–1575.

[67] C. Angelberger, D. Veynante, and F. Egolfopoulos. "LES of chemical and acoustic forcing of a premixed dump combustor". In: *Flow, Turbulence and Combustion* 65.2 (2000), pp. 205–222.

[68] P. Wolf et al. "Massively parallel LES of azimuthal thermo-acoustic instabilities in annular gas turbines". In: *Comptes Rendus Mecanique* 337.6-7 (2009), pp. 385–394.

[69] P. Wolf et al. "Acoustic and large eddy simulation studies of azimuthal modes in annular combustion chambers". In: *Combustion and Flame* 159.11 (2012), pp. 3398–3413.

[70] A. Ghani et al. "LES of longitudinal and transverse self-excited combustion instabilities in a bluff-body stabilized turbulent premixed flame". In: *Combustion and Flame* 162.11 (2015), pp. 4075–4083.

[71] T. Poinsot and D. Veynante. *Theoretical and numerical combustion*. RT Edwards Inc., 2005.

[72] A. Burcat and B. Ruscic. *Third millenium ideal gas and condensed phase thermochemical database for combustion with updates from active thermochemical tables*. Argonne National Laboratory Argonne, IL, 2005.

[73] R. J. Kee et al. "CHEMKIN-III: A FORTRAN chemical kinetics package for the analysis of gas-phase chemical and plasma kinetics". In: *Sandia national laboratories report SAND96-8216* (1996).

[74] S. B. Pope. *Turbulent flows*. 2001.

[75] L. F. Richardson. *Weather prediction by numerical process*. Cambridge University Press, 2007.

[76] J. Boussinesq. "Théorie de l'écoulement tourbillant". In: *Mem. Présentés par Divers Savants Acad. Sci. Inst. Fr* 23.46-50 (1877), pp. 6–5.

[77] J. Hinze. "Turbulence McGraw-Hill". In: *New York* 218 (1975).

[78] N. Peters. "Four lectures on turbulent combustion". In: *ERCOFTAC Summer School* (1997).

[79] N. Peters. *Turbulent combustion*. Cambridge university press, 2000.

[80] N. Peters. "Laninar flamelet concepts in turbulent combustion". In: *Symposium (International) on Conbustion*. Vol. 21. 1. Elsevier, 1988, pp. 1231–1250.

[81] N. Peters. "The turbulent burning velocity for large-scale and small-scale turbulence". In: *Journal of Fluid mechanics* 384 (1999), pp. 107–132.

[82] A. Leonard. "Energy cascade in large-eddy simulations of turbulent fluid flows". In: *Advances in geophysics* 18 (1975), pp. 237–248.

[83] M. P. Martin, U. Piomelli, and G. V. Candler. "Subgrid-scale models for compressible large-eddy simulations". In: *Theoretical and Computational Fluid Dynamics* 13.5 (2000), pp. 361–376.

[84] J. Smagorinsky. "General circulation experiments with the primitive equations: I. the basic experiment*". In: *Monthly weather review* 91.3 (1963), pp. 99–164.

[85] M. Lesieur and O. Metais. "New trends in large-eddy simulations of turbulence". In: *Annual review of fluid mechanics* 28.1 (1996), pp. 45–82.

[86] A. Vreman. "An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications". In: *Physics of fluids* 16.10 (2004), pp. 3670–3681.

[87] F. Nicoud et al. "Using singular values to build a subgrid-scale model for large eddy simulations". In: *Physics of Fluids* 23.8 (2011), p. 085106.

[88] M Germano. "A proposal for a redefinition of the turbulent stresses in the filtered Navier–Stokes equations". In: *The Physics of fluids* 29.7 (1986), pp. 2323–2324.

[89] M. Germano et al. "A dynamic subgrid-scale eddy viscosity model". In: *Physics of Fluids A: Fluid Dynamics (1989-1993)* 3.7 (1991), pp. 1760–1765.

[90] D. K. Lilly. "A proposed modification of the Germano subgrid-scale closure method". In: *Physics of Fluids A: Fluid Dynamics* 4.3 (1992), pp. 633–635.

[91] S. Ghosal et al. "A dynamic localization model for large-eddy simulation of turbulent flows". In: *Journal of Fluid Mechanics* 286 (1995), pp. 229–255.

[92] J. Bardina, J Ferziger, and W. Reynolds. "Improved subgrid-scale models for large-eddy simulation". In: *13th Fluid and PlasmaDynamics Conference*. 1980, p. 1357.

[93] S. Stolz and N. Adams. "An approximate deconvolution procedure for large-eddy simulation". In: *Physics of fluids* 11.7 (1999), pp. 1699–1701.

[94] Y. Zang, R. L. Street, and J. R. Koseff. "A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows". In: *Physics of fluids A: Fluid dynamics* 5.12 (1993), pp. 3186–3196.

[95] V. Armenio and U. Piomelli. "A Lagrangian mixed subgrid-scale model in generalized coordinates". In: *Flow, Turbulence and Combustion* 65.1 (2000), pp. 51–81.

[96] D. K. Lilly. *On the application of the eddy viscosity concept in the inertial subrange of turbulence*. National Center for Atmospheric Research, 1966.

[97] D. K. Lilly. "The representation of small scale turbulence in numerical simulation experiments". In: (1967).

[98] R. S. Rogallo and P. Moin. "Numerical simulation of turbulent flows". In: *Annual review of fluid mechanics* 16.1 (1984), pp. 99–137.

[99] C. Meneveau and J. Katz. "Scale-invariance and turbulence models for large-eddy simulation". In: *Annual Review of Fluid Mechanics* 32.1 (2000), pp. 1–32.

[100] D. Knight et al. "Large-Eddy Simulation of premixed turbulent combustion using flame surface density approach". In: *36th AIAA Aerospace Sciences Meeting and Exhibit*. 1998, p. 535.

[101] K. Bray, P. A. Libby, and J. Moss. "Unified modeling approach for premixed turbulent combustion Part I: General formulation". In: *Combustion and flame* 61.1 (1985), pp. 87–102.

[102] G. Lecocq et al. "Gradient and Counter-Gradient Modeling in Premixed Flames: Theoretical Study and Application to the LES of a Lean Premixed Turbulent Swirl-Burner". In: *Combustion Science and Technology* 182.4-6 (2010), pp. 465–479.

[103] T. Ma et al. "A posteriori testing of algebraic flame surface density models for LES". In: *Combustion Theory and Modelling* 17.3 (2013), pp. 431–482.

[104] C. Angelberger et al. "Large eddy simulations of combustion instabilities in premixed flames". In: *Proc. of the Summer Program*. 1998, pp. 61–82.

[105] M. Boger et al. "Direct numerical simulation analysis of flame surface density concept for large eddy simulation of turbulent premixed combustion". In: *Symposium (International) on Combustion*. Vol. 27. 1. Elsevier. 1998, pp. 917–925.

[106] J. B. Heywood. *Internal combustion engine fundamentals*. Vol. 930. McGraw-Hill New York, 1988.

[107] J. Wyngaard. "Atmospheric turbulence". In: *Annual Review of Fluid Mechanics* 24.1 (1992), pp. 205–234.

[108] A. Kempf, R. Lindstedt, and J. Janicka. "Large-eddy simulation of a bluff-body stabilized nonpremixed flame". In: *Combustion and flame* 144.1 (2006), pp. 170–189.

[109] S. M. Candel and T. J. Poinsot. "Flame stretch and the balance equation for the flame area". In: *Combustion Science and Technology* 70.1-3 (1990), pp. 1–15.

[110] L. Vervisch et al. "Surface density function in premixed turbulent combustion modeling, similarities between probability density function and flame surface approaches". In: *Physics of Fluids* 7.10 (1995), pp. 2496–2503.

[111] E. R. Hawkes. "Large eddy simulation of premixed turbulent combustion." PhD thesis. University of Cambridge, 2001.

[112] W. Lin. "Compressible large eddy simulation using unstructured grids". PhD thesis. University of Toronto, 2010.

[113] S. R. Turns. *An introduction to combustion. 2nd*. 2000.

[114] E. Hawkes and R. Cant. "Implications of a flame surface density approach to large eddy simulation of premixed turbulent combustion". In: *Combustion and Flame* 126.3 (2001), pp. 1617–1629.

[115] P. Yeung, S. Girimaji, and S. Pope. "Straining and scalar dissipation on material surfaces in turbulence: implications for flamelets". In: *Combustion and flame* 79.3-4 (1990), pp. 340–365.

[116] A. Trouvé. "The production of premixed flame surface area in turbulent shear flow". In: *Combustion and flame* 99.3-4 (1994), pp. 687–696.

[117] A. Kempf. *Large-eddy simulation of non-premixed turbulent flames*. VDI-Verlag, 2004.

[118] L. B. "A stable and accurate convective modelling procedure based on quadratic upstream interpolation". In: *Computer methods in applied mechanics and engineering* 19.1 (1979), pp. 59–98.

[119] H. K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson Education, 2007.

[120] G. Zhou. *Numerical simulations of physical discontinuities in single and multi-fluid flows for arbitrary Mach numbers*. Chalmers University of Technology, 1995.

[121] B. Van Leer. "Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme". In: *Journal of computational physics* 14.4 (1974), pp. 361–370.

[122] G. Van Albada, B. Van Leer, and W. Roberts Jr. "A comparative study of computational methods in cosmic gas dynamics". In: *Upwind and High-Resolution Schemes*. Springer, 1997, pp. 95–103.

[123] M. Kermani, A. Gerber, and J. Stockie. "Thermodynamically based moisture prediction using Roes scheme". In: *The 4th Conference of Iranian AeroSpace Society, Amir Kabir University of Technology, Tehran, Iran*. 2003.

[124] P. Roe. "Characteristic-based schemes for the Euler equations". In: *Annual review of fluid mechanics* 18.1 (1986), pp. 337–365.

[125] P. K. Sweby. "High resolution schemes using flux limiters for hyperbolic conservation laws". In: *SIAM journal on numerical analysis* 21.5 (1984), pp. 995–1011.

[126] F. S. Lien and M. Leschziner. "Upstream monotonic interpolation for scalar transport with application to complex turbulent flows". In: *International Journal for Numerical Methods in Fluids* 19.6 (1994), pp. 527–548.

[127] S. Chatkravathy. "High resolution applications of the Osher upwind scheme for the Euler equations, AIAA Paper 83-1943". In: *Proc. AIAA 6th Computational Fluid Dynamics Conference*. 1983, pp. 363–373.

[128] B. Koren. *A robust upwind discretization method for advection, diffusion and source terms*. Centrum voor Wiskunde en Informatica Amsterdam, 1993.

[129] R. Courant, K. Friedrichs, and H. Lewy. "Über die partiellen Differenzengleichungen der mathematischen Physik". In: *Mathematische annalen* 100.1 (1928), pp. 32–74.

[130] J. C. Butcher. "Coefficients for the study of Runge-Kutta integration processes". In: *Journal of the Australian Mathematical Society* 3.02 (1963), pp. 185–201.

[131] J. Crank and P. Nicolson. "A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type". In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Vol. 43. 01. Cambridge Univ Press. 1947, pp. 50–67.

[132] G. Hedstrom. "Nonreflecting boundary conditions for nonlinear hyperbolic systems". In: *Journal of computational Physics* 30.2 (1979), pp. 222–237.

[133] K. W. Thompson. "Time dependent boundary conditions for hyperbolic systems". In: *Journal of computational physics* 68.1 (1987), pp. 1–24.

[134] K. W. Thompson. "Time-dependent boundary conditions for hyperbolic systems, II". In: *Journal of Computational Physics* 89.2 (1990), pp. 439–461.

[135] M. B. Giles. "Nonreflecting boundary conditions for Euler equation calculations". In: *AIAA journal* 28.12 (1990), pp. 2050–2058.

[136] T. Poinsot and S. Lelef. "Boundary conditions for direct simulations of compressible viscous flows". In: *Journal of computational physics* 101.1 (1992), pp. 104–129.

[137] F. Nicoud. "Defining wave amplitude in characteristic boundary conditions". In: *Journal of Computational Physics* 149.2 (1999), pp. 418–422.

[138] C. S. Yoo et al. "Characteristic boundary conditions for direct simulations of turbulent counterflow flames". In: *Combustion Theory and Modelling* 9.4 (2005), pp. 617–646.

[139] C. S. Yoo and H. G. Im. "Characteristic boundary conditions for simulations of compressible reacting flows with multi-dimensional, viscous and reaction effects". In: *Combustion Theory and Modelling* 11.2 (2007), pp. 259–286.

[140] G. Lodato, P. Domingo, and L. Vervisch. "Three-dimensional boundary conditions for direct and large-eddy simulation of compressible viscous flows". In: *Journal of Computational Physics* 227.10 (2008), pp. 5105–5143.

[141] E. Baum et al. "On the validation of LES applied to internal combustion engine flows: Part 1: comprehensive experimental database". In: *Flow, Turbulence and Combustion* (2013), pp. 1–29.

[142] P. Janas et al. "On the evolution of the flow field in a spark ignition engine". In: *Flow, Turbulence and Combustion* (2016), pp. 1–28.

[143] P. Min. *A 3D Mesh Voxelizer*. http://www.cs.princeton.edu/~min/binvox/. Accessed: 2017-02-20.

[144] R. Mittal and G. Iaccarino. "Immersed boundary methods". In: *Annual Review of Fluid Mechanics* 37 (2005), pp. 239–261.

[145] H. Forrer and M. Berger. "Flow simulations on Cartesian grids involving complex moving geometries". In: *Hyperbolic problems: theory, numerics, applications*. Springer, 1999, pp. 315–324.

[146] G. Bruschi et al. *Drag coefficient of a cylinder*. 2003.

[147] A. Roshko. "Experiments on the flow past a circular cylinder at very high Reynolds number". In: *Journal of Fluid Mechanics* 10.03 (1961), pp. 345–356.

[148] M. Malik and D. Poll. "Effect of curvature on three-dimensional boundary-layer stability". In: *AIAA journal* 23.9 (1985), pp. 1362–1369.

[149] N. K. Delany and N. E. Sorensen. "Low-speed drag of cylinders of various shapes". In: (1953).

[150] P. Catalano et al. "Numerical simulation of the flow around a circular cylinder at high Reynolds numbers". In: *International Journal of Heat and Fluid Flow* 24.4 (2003), pp. 463–469.

[151] M. C. Lai and C. S. Peskin. "An immersed boundary method with formal second-order accuracy and reduced numerical viscosity". In: *Journal of Computational Physics* 160.2 (2000), pp. 705–719.

[152] U. Piomelli and E. Balaras. "Wall-layer models for large-eddy simulations". In: *Annual review of fluid mechanics* 34.1 (2002), pp. 349–374.

[153] M. Wang and P. Moin. "Dynamic wall modeling for large-eddy simulation of complex turbulent flows". In: *Physics of Fluids (1994-present)* 14.7 (2002), pp. 2043–2051.

[154] R. D. Moser, J. Kim, and N. N. Mansour. "Direct numerical simulation of turbulent channel flow up to Re= 590". In: *Phys. Fluids* 11.4 (1999), pp. 943–945.

[155] A. Morse, J. Whitelaw, and M. Yianneskis. "Turbulent flow measurements by laser doppler anemometry in a motored reciprocating engine". In: *Imperial College Mechanical Engineering Department Report FS/78/24* (1978).

[156] T. Nguyen et al. "Large eddy simulation of an internal combustion engine using an efficient immersed boundary technique". In: *Flow, Turbulence and Combustion* 97.1 (2016), pp. 191–230.

[157] T. Nguyen et al. "LES of flow processes in an SI engine using two approaches: OpenFoam and PsiPhi". In: *SAE Technical Paper 2014-01-1121* (2014).

[158] M. Baumann, F. di Mare, and J. Janicka. "On the validation of large eddy simulation applied to internal combustion engine flows part II: numerical analysis". In: *Flow, Turbulence and Combustion* 92.1-2 (2014), pp. 299–317.

[159] H. M. Ismail, H. K. Ng, and S. Gan. "Evaluation of non-premixed combustion and fuel spray models for in-cylinder diesel engine simulation". In: *Applied Energy* 90.1 (2012), pp. 271–279.

[160] K. Sone and S. Menon. "Effect of subgrid modeling on the in-cylinder unsteady mixing process in a direct injection engine". In: *Transactions-american society of mechanical engineers journal of engineering for gas turbines and power* 125.2 (2003), pp. 435–443.

[161] OpenFoam. *OpenCFD release OpenFOAM*. http://www.openfoam.com. Accessed: 2017-03-01.

[162] A. Kempf, B. Geurts, and J. Oefelein. "Error analysis of large-eddy simulation of the turbulent non-premixed sydney bluff-body flame". In: *Combustion and Flame* 158.12 (2011), pp. 2408–2419.

[163] T. Nguyen and A. Kempf. "Large eddy simulation of an IC engine: An approach for moving boundaries in IC engine simulations". In: *Proc. of the European Combustion Institute*. Vol. 6. 2013.

[164] C. S. Peskin. "Flow patterns around heart valves: a numerical method". In: *Journal of Computational Physics* 10.2 (1972), pp. 252–271.

[165] P. Janas et al. "Numerical simulation of flame front propagation in a spark ignition engine". In: *Proc. of the European Combustion Institute*. Vol. 6. 2013.

[166] T. Lucchini et al. "Full-cycle CFD modeling of air/fuel mixing process in an optically accessible GDI engine". In: *SAE International Journal of Engines* 6.2013-24-0024 (2013), pp. 1610–1625.

[167] F. Contino et al. "Simulations of advanced combustion modes using detailed chemistry combined with tabulation and mechanism reduction techniques". In: *SAE International Journal of Engines* 5.2012-01-0145 (2012), pp. 185–196.

[168] T. Lucchini et al. "Numerical investigation of non-reacting and reacting diesel sprays in constant-volume vessels". In: *SAE International Journal of Fuels and Lubricants* 2.2009-01-1971 (2009), pp. 966–975.

[169] T. Lucchini et al. *Development of a CFD approach to model fuel-air mixing in gasoline direct-injection engines*. Tech. rep. SAE Technical Paper, 2012.

[170] H. Jasak and Z. Tukovic. "Automatic mesh motion for the unstructured finite volume method". In: *Transactions of FAMENA* 30.2 (2006), pp. 1–20.

[171] R. I. Issa. "Solution of the implicitly discretised fluid flow equations by operator-splitting". In: *Journal of computational physics* 62.1 (1986), pp. 40–65.

[172] L. G. Marcantoni, J. Tamagno, and S. Elaskar. "High speed flow simulation using OpenFoam". In: *Mecánica Computacional* 31 (2012), pp. 2939–2959.

[173] M. Pettit et al. "Large-Eddy Simulation and experiments on non-premixed highly turbulent opposed jet flows". In: *Proceedings of the Combustion Institute* 33.1 (2011), pp. 1391–1399.

[174] C. Olbricht et al. "LES of lifted flames in a gas turbine model combustor using top-hat filtered PFGM chemistry". In: *Fuel* 96 (2012), pp. 100–107.

[175] A. Rittler, F. Proch, and A. Kempf. "Large eddy simulation of the Sydney piloted spray burner". In: *Proc. of the European Combustion Institute*. Vol. 6. 2013.

[176] B. Franchetti et al. "Large eddy simulation of a pulverised coal jet flame". In: *Proceedings of the Combustion Institute* 34.2 (2013), pp. 2419–2426.

[177] G. Roth and J. Katz. "Five techniques for increasing the speed and accuracy of PIV interrogation". In: *Measurement Science and Technology* 12.3 (2001), p. 238.

[178] C. D. Pierce and P. Moin. "Progress-variable approach for large-eddy simulation of non-premixed turbulent combustion". In: *Journal of Fluid Mechanics* 504 (2004), pp. 73–97.

[179] H. Pitsch. "Large-eddy simulation of turbulent combustion". In: *Annual Review of Fluid Mechanics* 38 (2006), pp. 453–482.

[180] A. Kempf, M. Klein, and J. Janicka. "Efficient generation of initial-and inflow-conditions for transient turbulent flows in arbitrary geometries". In: *Flow, Turbulence and combustion* 74.1 (2005), pp. 67–84.

[181] C. Fureby. "A fractal flame-wrinkling large eddy simulation model for premixed turbulent combustion". In: *Proceedings of the Combustion Institute* 30.1 (2005), pp. 593 –601. ISSN: 1540-7489.

[182] W. W. Kim, S. Menon, and H. C. Mongia. "Large-eddy simulation of a gas turbine combustor flow". In: *Combustion Science and Technology* 143.1-6 (1999), pp. 25–62.

[183] M. Boileau et al. "LES of an ignition sequence in a gas turbine engine". In: *Combustion and Flame* 154.1 (2008), pp. 2–22.

[184] F. Di Mare, W. Jones, and K. Menzies. "Large eddy simulation of a model gas turbine combustor". In: *Combustion and Flame* 137.3 (2004), pp. 278–294.

[185] S. V. Apte, M. Gorokhovski, and P. Moin. "LES of atomizing spray with stochastic modeling of secondary breakup". In: *International Journal of Multiphase Flow* 29.9 (2003), pp. 1503–1522.

[186] V. Sankaran and S. Menon. "LES of spray combustion in swirling flows*". In: *Journal of Turbulence* 3.11 (2002), pp. 1–23.

[187] A. Rittler, F. Proch, and A. M. Kempf. "LES of the Sydney piloted spray flame series with the PFGM/ATF approach and different sub-filter models". In: *Combustion and Flame* 162.4 (2015), pp. 1575–1598.

[188] P. Edge et al. "LES modelling of air and oxy-fuel pulverised coal combustion - impact on flame properties". In: *Proceedings of the Combustion Institute* 33.2 (2011), pp. 2709–2716.

[189] H. Zhou, G. Flamant, and D. Gauthier. "DEM-LES simulation of coal combustion in a bubbling fluidized bed Part II: coal combustion at the particle level". In: *Chemical Engineering Science* 59.20 (2004), pp. 4205–4215.

[190] D. Haworth and K. Jansen. "Large-eddy simulation on unstructured deforming meshes: towards reciprocating IC engines". In: *Computers & fluids* 29.5 (2000), pp. 493–524.

[191] E. Fadlun et al. "Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations". In: *Journal of Computational Physics* 161.1 (2000), pp. 35–60.

[192] K. Sone, N. Patel, and S. Menon. "Large eddy simulation of fuel-air mixing in an internal combustion engine". In: *AIAA paper* 635 (2001).

[193] H. W. Wu and S. W. Perng. "LES analysis of turbulent flow and heat transfer in motored engines with various SGS models". In: *International journal of heat and mass transfer* 45.11 (2002), pp. 2315–2328.

[194] D. Goryntsev et al. "Large eddy simulation based analysis of the effects of cycle-to-cycle variations on air–fuel mixing in realistic DISI IC-engines". In: *Proceedings of the Combustion Institute* 32.2 (2009), pp. 2759–2766.

[195] D. Goldstein, R. Handler, and L. Sirovich. "Direct numerical simulation of turbulent flow over a modeled riblet covered surface". In: *Journal of Fluid Mechanics* 302 (1995), pp. 333–376.

[196] R. Verzicco et al. "LES in complex geometries using boundary body forces". In: *Center for Turbulence Research Proceedings of the Summer Program, NASA Ames and Stanford University* (1998), pp. 171–186.

[197] E. Balaras. "Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations". In: *Computers & Fluids* 33.3 (2004), pp. 375–404.

[198] J. Mohd Yusof. "Combined immersed-boundary/B-spline methods for simulations of ow in complex geometries". In: *Annual Research Briefs. NASA Ames Research Center, Stanford University Center of Turbulence Research: Stanford* (1997), pp. 317–327.

[199] M. D. de Tullio et al. "An immersed boundary method for compressible flows using local grid refinement". In: *Journal of Computational Physics* 225.2 (2007), pp. 2098–2117.

[200] K. Taira and T. Colonius. "The immersed boundary method: a projection approach". In: *Journal of Computational Physics* 225.2 (2007), pp. 2118–2137.

[201] H. Udaykumar, W. Shyy, and M. Rao. "Elafint: a mixed Eulerian–Lagrangian method for fluid flows with complex and moving boundaries". In: *International journal for numerical methods in fluids* 22.8 (1996), pp. 691–712.

[202] V. Mittal et al. “LES of Gas Exchange in IC Engines”. In: *Oil & Gas Science and Technology–Revue d'Energies nouvelles* 69.1 (2014), pp. 29–40.

[203] C. Hirt, A. A. Amsden, and J. Cook. “An arbitrary Lagrangian-Eulerian computing method for all flow speeds”. In: *Journal of Computational Physics* 14.3 (1974), pp. 227–253.

[204] H. Udaykumar et al. “A sharp interface Cartesian grid method for simulating flows with complex moving boundaries”. In: *Journal of Computational Physics* 174.1 (2001), pp. 345–380.

[205] T. Fansler, M. Drake, and B. Böhm. “High-speed Mie-scattering diagnostics for spray-guided gasoline engine development”. In: *Proceedings of the 8th International Symposium on Combustion Diagnostics, Baden-Baden*. 2008, pp. 413–425.

[206] N. Gourdain et al. “High performance parallel computing of flows in complex geometries: I. methods”. In: *Computational Science & Discovery* 2.1 (2009), p. 015003.

[207] V. Moureau, P. Domingo, and L. Vervisch. “From Large-Eddy Simulation to Direct Numerical Simulation of a lean premixed swirl flame: Filtered laminar flame-PDF modeling”. In: *Combustion and Flame* 158.7 (2011), pp. 1340–1357.

[208] M. Rieth et al. “Comparison of the Sigma and Smagorinsky LES models for grid generated turbulence and a channel flow”. In: *Computers & Fluids* 99 (2014), pp. 172–181.

[209] I. Celik, Z. Cehreli, and I. Yavuz. “Index of resolution quality for large eddy simulations”. In: *Journal of Fluids Engineering* 127.5 (2005), pp. 949–958.

[210] X. Luo et al. “In-cylinder wall temperature influence on unburned hydrocarbon emissions during transitional period in an optical engine using a laser-induced phosphorescence technique”. In: *SAE International Journal of Engines* 7.2014-01-1373 (2014), pp. 995–1002.

[211] C. Wilhelmsson et al. “Combustion chamber wall temperature measurement and modeling during transient HCCI operation”. In: *SAE Technical Papers 2005-01-3731* (2005).

[212] D. Cleary and G. Silvas. “Unthrottled engine operation with variable intake valve lift, duration, and timing”. In: *SAE Technical Paper 2007-01-1282* (2007).

[213] G. Davis, R. Tabaczynski, and R. Belaire. "The effect of intake valve lift on turbulence intensity and burnrate in SI engines-model versus experiment". In: *SAE Technical Paper 840030* (1984).

[214] P. Kreuter, P. Heuser, and M. Schebitz. "Strategies to improve SI-engine performance by means of variable intake lift, timing and duration". In: *SAE Technical Paper 920449* (1992).

[215] S. Ghosal and P. Moin. "The basic equations for the large eddy simulation of turbulent flows in complex geometry". In: *Journal of Computational Physics* 118.1 (1995), pp. 24–37.

[216] R. Gerdes, C. Köberle, and J. Willebrand. "The influence of numerical advection schemes on the results of ocean general circulation models". In: *Climate Dynamics* 5.4 (1991), pp. 211–226.

[217] A. Nakayama and S. Vengadesan. "On the influence of numerical schemes and subgrid–stress models on large eddy simulation of turbulent flow past a square cylinder". In: *International journal for numerical methods in fluids* 38.3 (2002), pp. 227–253.

[218] J. Ekaterinaris. "High-order accurate, low numerical diffusion methods for aerodynamics". In: *Progress in Aerospace Sciences* 41.3 (2005), pp. 192–300.

[219] H. Yee, N. Sandham, and M. Djomehri. "Low-dissipative high-order shock-capturing methods using characteristic-based filters". In: *Journal of computational physics* 150.1 (1999), pp. 199–238.

[220] H. Jasak. *Error analysis and estimation for finite volume method with applications to fluid flow*. Tech. rep. 1996.

[221] A. Kravchenko and P. Moin. "On the effect of numerical errors in large eddy simulations of turbulent flows". In: *Journal of Computational Physics* 131.2 (1997), pp. 310–322.

[222] J. Aubin, D. Fletcher, and C. Xuereb. "Modeling turbulent flow in stirred tanks with CFD: the influence of the modeling approach, turbulence model and numerical scheme". In: *Experimental thermal and fluid science* 28.5 (2004), pp. 431–445.

[223] D. Chock and A. Dunker. "A comparison of numerical methods for solving the advection equation". In: *Atmospheric Environment (1967)* 17.1 (1983), pp. 11–24.

[224] A. Misdariis et al. "Numerical methods and turbulence modeling for LES of piston engines: impact on flow motion and combustion". In: *Oil & Gas Science and Technology Journal* 69 (2014), pp–83.

[225] F. di Mare, R. Knappstein, and M. Baumann. "Application of LES-quality criteria to internal combustion engine flows". In: *Computers & Fluids* 89 (2014), pp. 200–213.

[226] M. Keating. "Accelerating CFD solutions". In: *advantage* 1 (2011), p. 48.

[227] M. Grigoriev, C. Swiatek, and J. Hitt. "Benchmarking CD-adapcos STAR-CCM+ in a Production Design Environment". In: *ASME Turbo Expo 2010: Power for Land, Sea, and Air*. American Society of Mechanical Engineers. 2010, pp. 1019–1025.

[228] H. Jasak, A. Jemcov, Z. Tukovic, et al. "OpenFOAM: A C++ library for complex physics simulations". In: *International workshop on coupled methods in numerical dynamics*. Vol. 1000. IUC Dubrovnik, Croatia. 2007, pp. 1–20.

[229] J. Keskinen. "Large eddy simulation of in-cylinder flows". In: (2016).

[230] A. Wehrfritz et al. "Large eddy simulation of high-velocity fuel sprays: studying mesh resolution and breakup model effects for Spray A". In: *Atomization and Sprays* 23.5 (2013).

[231] J. Boris et al. "New insights into large eddy simulation". In: *Fluid dynamics research* 10.4-6 (1992), pp. 199–228.

[232] S. Breuer, M. Oberlack, and N. Peters. "Non-isotropic length scales during the compression stroke of a motored piston engine". In: *Flow, turbulence and combustion* 74.2 (2005), pp. 145–167.

[233] M. Klein. "An attempt to assess the quality of large eddy simulations in the context of implicit filtering". In: *Flow, Turbulence and Combustion* 75.1-4 (2005), pp. 131–147.

[234] P. Gousseau, B. Blocken, and B. Van Heijst. "Quality assessment of large-eddy simulation of wind flow around a high-rise building: validation and solution verification". In: *Computers & Fluids* 79 (2013), pp. 120–133.

[235] M. Freitag and M. Klein. "An improved method to assess the quality of large eddy simulations in the context of implicit filtering". In: *Journal of Turbulence* 7 (2006), N40.

[236] Y. Addad et al. "Optimal unstructured meshing for large eddy simulations". In: *Quality and Reliability of Large-Eddy Simulations*. Springer, 2008, pp. 93–103.

[237] A. Yoshizawa. "Statistical theory for compressible turbulent shear flows, with the application to subgrid modeling". In: *Physics of Fluids (1958-1988)* 29.7 (1986), pp. 2152–2164.

[238] J. J. Quirk. *A cartesian grid approach with hierarchical refinement for compressible flows*. Tech. rep. DTIC Document, 1994.

[239] M. J. Berger and P. Colella. "Local adaptive mesh refinement for shock hydrodynamics". In: *Journal of computational Physics* 82.1 (1989), pp. 64–84.